オペレーティング システム問題のトラブルシューティング

このオペレーティング システムの問題のトラブルシューティングでは、次のトピックを扱います。

- オペレーティング システムのトラブルシューティング, on page 1

オペレーティング システムのトラブルシューティング

このセクションは、システムのオペレーティング システムに関する問題のトラブルシュートに役立ちます。

注： 問題が解決しない場合は、Wistron テクニカル サポートにお問い合わせください。

トピック：

- 仮想メディアを介したオペレーティング システムのインストール方法（Ubuntu 22.04）
- AMD GPU
  - Linuxで amdgpu ドライバをブラックリストに追加する方法
  - Ubuntu で AMD MI300X ROCm およびドライバをインストールおよび再インストールする方法
  - AMDGPU ドライバにより、AMD プラットフォームで致命的なエラーが発生する
  - ROCm のアンインストール方法
  - マルチ GPU システムで AMD アプリケーションがハングするのはなぜですか
  - BMCコンソールを介して AMD MI300X UBB ファームウェアを更新する方法

仮想メディアを使用したオペレーティングシステムのインストール方法 - Ubuntu 22.04

はじめる前に

1. インストールするオペレーティング システムをサーバがサポートしていることを確認してください。
2. サーバで使用可能なOSソース メディアがあることを確認します。
3. 必要なインストール ファイルを格納できる保管場所を用意します。

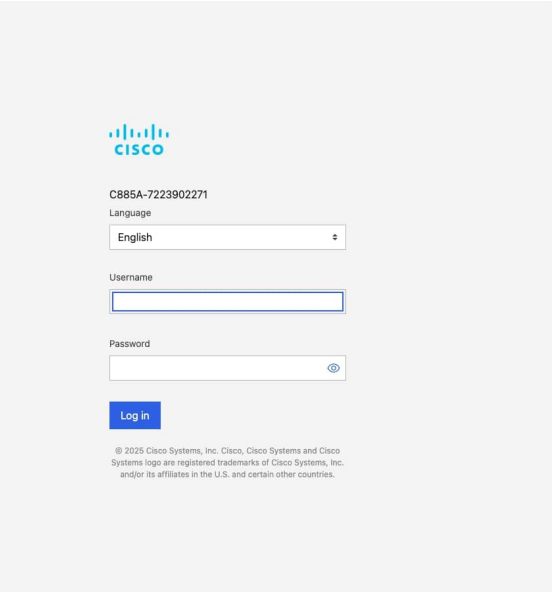
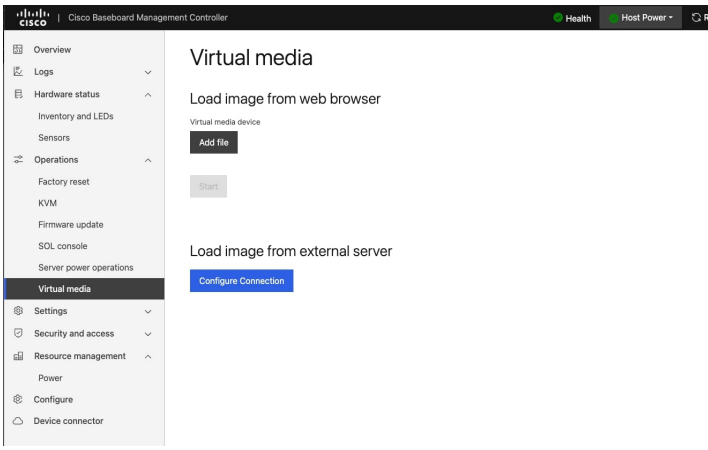
SUMMARY STEPS

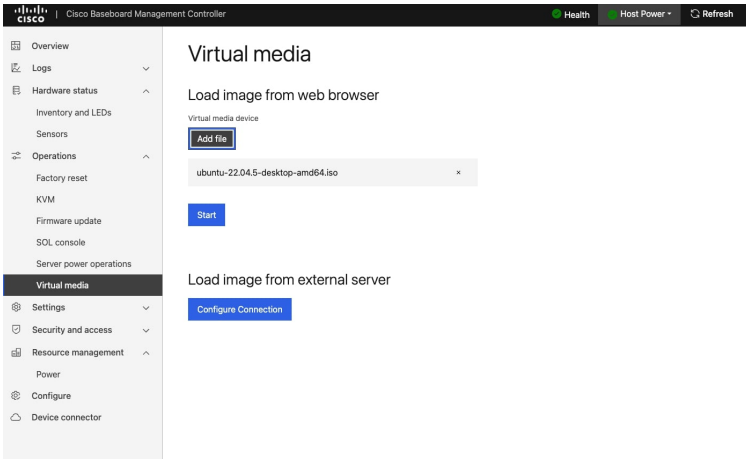
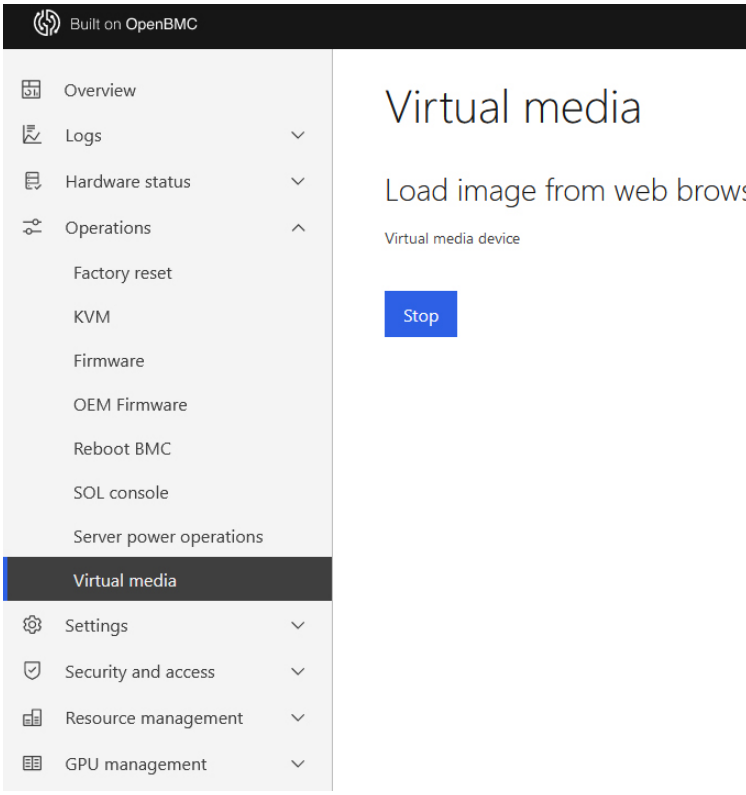
1. Web ブラウザでサーバの IP アドレスを入力して、OpenBMC Web UI インターフェイスにアクセスします。
2. OS ソース メディアを仮想メディアドライブにマウントします。
3. 以下の **[ファイルの追加（Add file）]**（ ISOファイルなど） を使用して、リモート対応メディア デバイスを追加します。
4. 以下の **[開始（Start）]** をクリックして、ファイルの追加後に仮想メディアを開始します。
5. ISO イメージのマウントが成功した後。
6. 変更をパスワードを変更、サーバを再起動して、 OSインストール イメージをブートします。
7. BIOS セットアップ電力事業またはブート メニューで、 **[保存して終了（Save & Exit）]** セクションまたは同様のオプションに移動します。
8. **UEFI**： **Linux File-CD Gadget 0515** または使用可能なブート デバイスから同様のエントリを選択します。
9. 変更を保存し、BIOSセットアップユーティリティまたはブートメニューを終了します。
10. サーバは、選択されたブート デバイス（リモート対応メディア ドライブ上のOSインストール イメージ）からブートします。
11. インストールが完了すると、サーバは再度リブートします。今回は、サーバのローカルディスクにインストールされているオペレーティング システムからブートします。

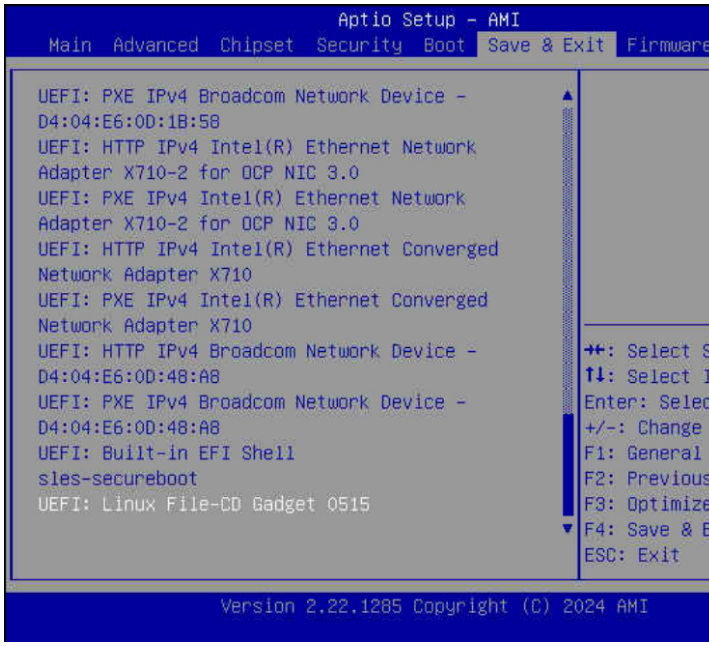
DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	Web ブラウザでサーバの IP アドレスを入力して、OpenBMC Web UI インターフェイスにアクセスします。	**Note** デフォルトのユーザー名は、 **root** で、パスワードは **0penBmc**です。

	Command or Action	Purpose
		Figure 1: BMC WebGUI へのログイン 
ステップ 2	OS ソース メディアを仮想メディア ドライブにマウントします。	
ステップ 3	以下の **[ファイルの追加（Add file）]**（ISOファイルなど）を使用して、リモート対応メディア デバイスを追加します。	***Figure 2:*** メディア ファイルの追加 

	Command or Action	Purpose
ステップ 4	以下の **[開始（Start）]** をクリックして、ファイルの追加後に仮想メディアを開始します。	***Figure 3: ISO*** イメージの起動 
ステップ 5	ISO イメージのマウントが成功した後。	***Figure 4: ISO*** のマウント 
ステップ 6	変更をパスワードを変更、サーバを再起動して、OSインストール イメージをブートします。	

	Command or Action	Purpose
ステップ 7	BIOSセットアップ電力事業またはブート メニューで、**[保存して終了（Save & Exit）]** セクションまたは同様のオプションに移動します。	
ステップ 8	**UEFI**： **Linux File-CD Gadget 0515** または使用可能なブート デバイスから同様のエントリを選択します。	
ステップ 9	変更を保存し、BIOSセットアップユーティリティまたはブート メニューを終了します。	
ステップ 10	サーバは、選択されたブート デバイス（リモート対応メディア ドライブ上のOSインストールイメージ）からブートします。	***Figure 5: BIOS*** **からの仮想メディアの起動** 
ステップ 11	インストールが完了すると、サーバは再度リブートします。今回は、サーバのローカル ディスクにインストールされているオペレーティング システムからブートします。	

AMD GPU

Linuxで amdgpu ドライバをブラックリストに追加する方法

原因： amdgpu ドライバまたはファームウェアに問題がある場合、ユーザーが amdgpu ドライバをロードしたときに、システムがシャットダウンまたは再起動する可能性があります。そのため、ユーザーは amdgpu ドライバをブラックリストて、自動的にロードされないようにする

必要があります。ユーザーがブラックリストの設定を忘れた場合、問題が解消されない場合があります。

解決方法：

次の **GRUB_CMDLINE_LINUX** （ **/etc/default/grub**内）の後ろのパラメータを追加するには、次の手順を実行します。

SUMMARY STEPS

1. 次のコマンドを実行して、テキスト エディタで GRUB 構成ファイルを開きます。 `$ sudo nano /etc/default/grub`
2. ファイルで **GRUB_CMDLINE_LINUX** から始まる行を見つけます。
3. `Ctrl + X`を押して変更保存し、、その後に `Y`を入力し、 `Enter`を押します。
4. 次のコマンドを実行して、GRUB 構成を更新します。
5. 変更を有効にするには、システムをリブートします。
6. amdgpu がブラックリストに追加されている場合は、次のコマンドを使用してモジュールを手動でロードする必要があります。

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	次のコマンドを実行して、テキスト エディタで GRUB 構成ファイルを開きます。 `$ sudo nano /etc/default/grub`	
ステップ 2	ファイルで **GRUB_CMDLINE_LINUX** から始まる行を見つけます。	たとえば、パラメータ「ブラックリスト=amdgpu iommu=pt pci=noats」を追加する必要がある場合、行は次のようになります。 `GRUB_CMDLINE_LINUX=”modprobe.blacklist=amdgpu iommu=pt pci=noats”`
ステップ 3	`Ctrl + X`を押して変更保存し、、その後に `Y`を入力し、 `Enter`を押します。	
ステップ 4	次のコマンドを実行して、GRUB 構成を更新します。	`$sudo grub-mkconfig -o /boot/efi/EFI/ubuntu/grub.cfg$sudo update-grub`
ステップ 5	変更を有効にするには、システムをリブートします。	`$sudo reboot`
ステップ 6	amdgpuがブラックリストに追加されている場合は、次のコマンドを使用してモジュールを手動でロードする必要があります。	`$modprobe amdgpu`

What to do next

注：再起動後、amdgpu ドライバに指定されたパラメータまたはオプションがロードされます。ドライバを手動でロードすると、問題や競合が発生する可能性があることに注意してください。問題が発生した場合は、ステップ 2 で追加したパラメータまたはオプションを確認するか、AMD のテクニカル サポートに問い合わせてください。

Ubuntu での AMD MI300X ROCm およびドライバのインストール方法

インストール前提条件

ROCm をインストールする前に、次の前提条件を満たします。

SUMMARY STEPS

1. システムでサポートされている Linux バージョンが存在することを確認します。
2. カーネル バージョンを確認します。

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	システムでサポートされている Linux バージョンが存在することを確認します。	• Linux ディストリビューションの情報を取得するには、次のコマンドを使用します。
		`$ uname -m && cat /etc/*release`
		• Linux ディストリビューションが サポートされている分散に一致していることを確認します。
		例： Ubuntu システムで上記のコマンドを実行すると、次の出力が生成されます。
		`x86_64`
		`DISTRIB_ID=Ubuntu`
		`DISTRIB_RELEASE=20.04`
		`DISTRIB_CODENAME=focal`
		`DISTRIB_DESCRIPTION="Ubuntu 20.04.5 LTS"`

ステップ 2	カーネル バージョンを確認します。	

What to do next

• Linuxシステムのカーネル バージョンを確認するには、次のコマンドを入力します。

```
$uname -srmv
```

例： 上記のコマンドは、カーネル バージョンを次の形式でリストします。

```
Linux 5.15.0-46-generic #44~20.04.5-Ubuntu SMP Fri Jun 24
13:27:29 UTC 2022 x86_64
```

- 次の サポートされるオペレーティング システムに記載されているように、カーネル バージョンがシステム要件を満たしていることを確認します。
- **ROCm のインストール方法**：

  Linuxでの ROCm のインストールに関する簡単な概要については、推奨のオペレーティング システムとインストール方法を選択し、表に記載されている手順に従ってください。詳細なインストール手順については、 インストール オプションを参照してください。

方法 1：

- **Ubuntu 22.04**

  パッケージ マネージャ

```
sudo apt install "linux-headers-$(uname -r)"
"linux-modules-extra-$(uname -r)"sudo usermod -a -G render,video
$LOGNAME # Adding current user to Video, Render groups. See
prerequisites.wget
https://repo.radeon.com/amdgpu-install/6.1.1/ubuntu/jammy/amdgpu-install_6.1.60101-1_all.debsudo
apt install ./amdgpu-install_6.1.60101-1_all.debsudo apt updatesudo
apt install amdgpu-dkmssudo apt install rocmecho "Please reboot
system for all settings to take effect."
```

- **AMDGPU をインストールする方法**：

  AMDGPU インストーラ

```
sudo apt install "linux-headers-$(uname -r)"
"linux-modules-extra-$(uname -r)"sudo usermod -a -G render,video
$LOGNAME # Adding current user to Video, Render groups. See
prerequisites.sudo apt updatewget
https://repo.radeon.com/amdgpu-install/6.1.1/ubuntu/jammy/amdgpu-install_6.1.60101-1_all.debsudo
apt install ./amdgpu-install_6.1.60101-1_all.debsudo amdgpu-install
--usecase=graphics,rocm
```

方法 2：

1. ROCmをインストールする前に、オペレーティング システムがクリーンであることを確認するか、古いバージョンの ROCm をアンインストールしてください。
2. 次の **uname -r** コマンドを実行して、カーネル バージョンを確認し、ROCm の README またはドキュメントで指定されている要件を満たしていることを確認します。
3. 次のコマンドを実行して ROCm パッケージを抽出します。

   ```
   $ tar -xvzf $ROCm_FILE
   ```

   次の **$ ROCm_FILE** をダウンロードした ROCm ファイルの名前に置き換えます。
4. 次のコマンドを実行して、ROCmディレクトリに変更します。

   ```
   $ cd $ROCm_DIR
   ```

   次の **$ROCm_DIR** を、抽出された ROCm ディレクトリの名前に置き換えます。
5. 次のコマンドを実行して、amdgpu-install スクリプトに実行権限を付与します。

   ```
   $chmod u+x amdgpu-install
   ```

次のコマンドを実行して、インストール プロセスを開始します。

```
$./amdgpu-install --usecase=hiplibsdk,rocm
```

プロンプトに従って、インストール プロセス中に必要な入力を行います。

ROCm のインストールが完了したら、システムを再起動する前に、amdgpu ドライバをブラックリストして、オペレーティング システムの起動時の自動プローブを防ぐことを推奨します。

amdgpu ドライバのブラックリストに追加する方法に関する詳細は、 **Ubuntu で amdgpu をブラックリストに追加する方法** セクションを参照してください。

提供される手順は一般的なガイドラインであり、正確なコマンドまたは手順は、特定のOSやROCmバージョンによって異なる場合があることに注意してください。お使いのセットアップに固有の詳細な手順については、ROCmのドキュメントまたは公式リソースを参照してください。

AMDGPU ドライバにより、AMD プラットフォームで致命的なエラーが発生する

原因： amdgpu ドライバの一部のバージョンには、AMD プラットフォームで致命的なエラーを引き起こす可能性のある問題が発生する場合があります。

解決方法：amdgpu ドライバが原因で発生する可能性のある致命的なエラーを回避するために、次の 2 つの方法を試してください。

方法1: Linux で amdgpu ドライバをブラックリストに追加します

1. 次のコマンドを実行して、テキスト エディタで GRUB 構成ファイルを開きます。

   $ **sudo nano /etc/default/grub**

2. ファイル内で「GRUB_CMDLINE_LINUX」で始まる行を見つけます。

   次の “**modprobe.blacklist=amdgpu**” **"iommu=pt"**と**"pci=noats"** パラメータを、引用符（""）内の既存コンテンツの後に追加します。

   たとえば、次のような記述があります。

   $ **GRUB_CMDLINE_LINUX=" modprobe.blacklist=amdgpu iommu=pt pci=noats "**

3. Ctrl + X、Y、最後に Enter を押して変更をパスワード を 変更。

4. 次のコマンドを実行して、GRUB 構成を更新します。

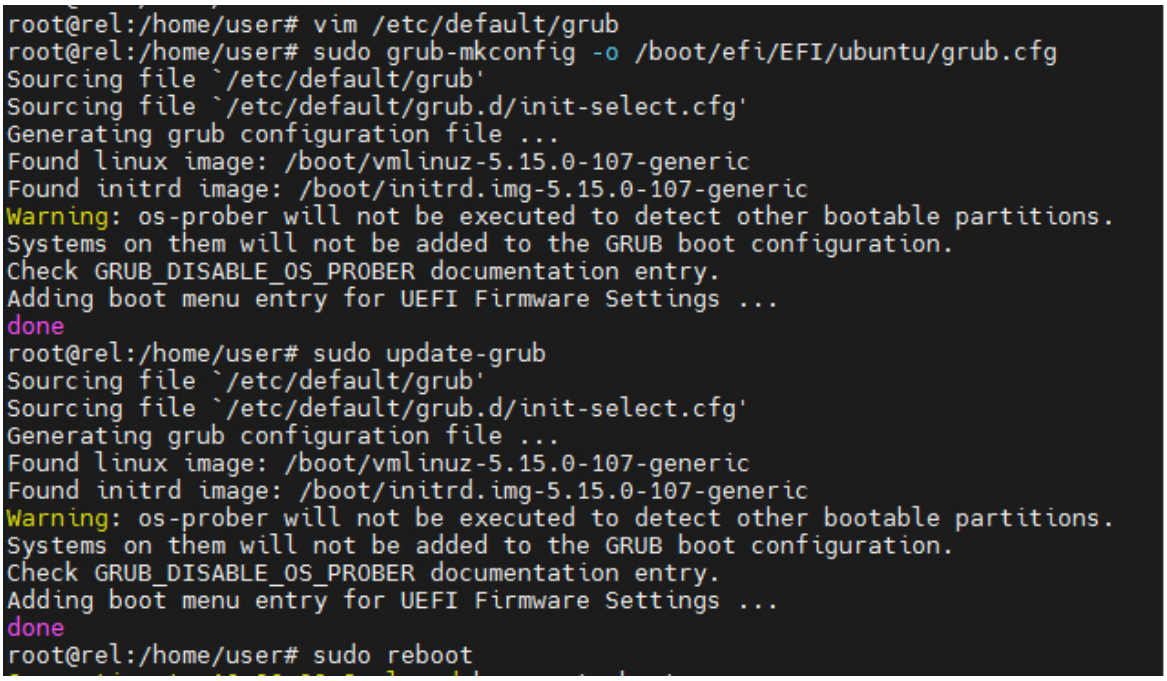
   $sudo grub-mkconfig -o /boot/efi/EFI/ubuntu/grub.cfg

   $sudo update-grub

5. 変更を有効にするには、システムをリブートします。

   $sudo reboot

Figure 6: GRUB 構成を更新した後



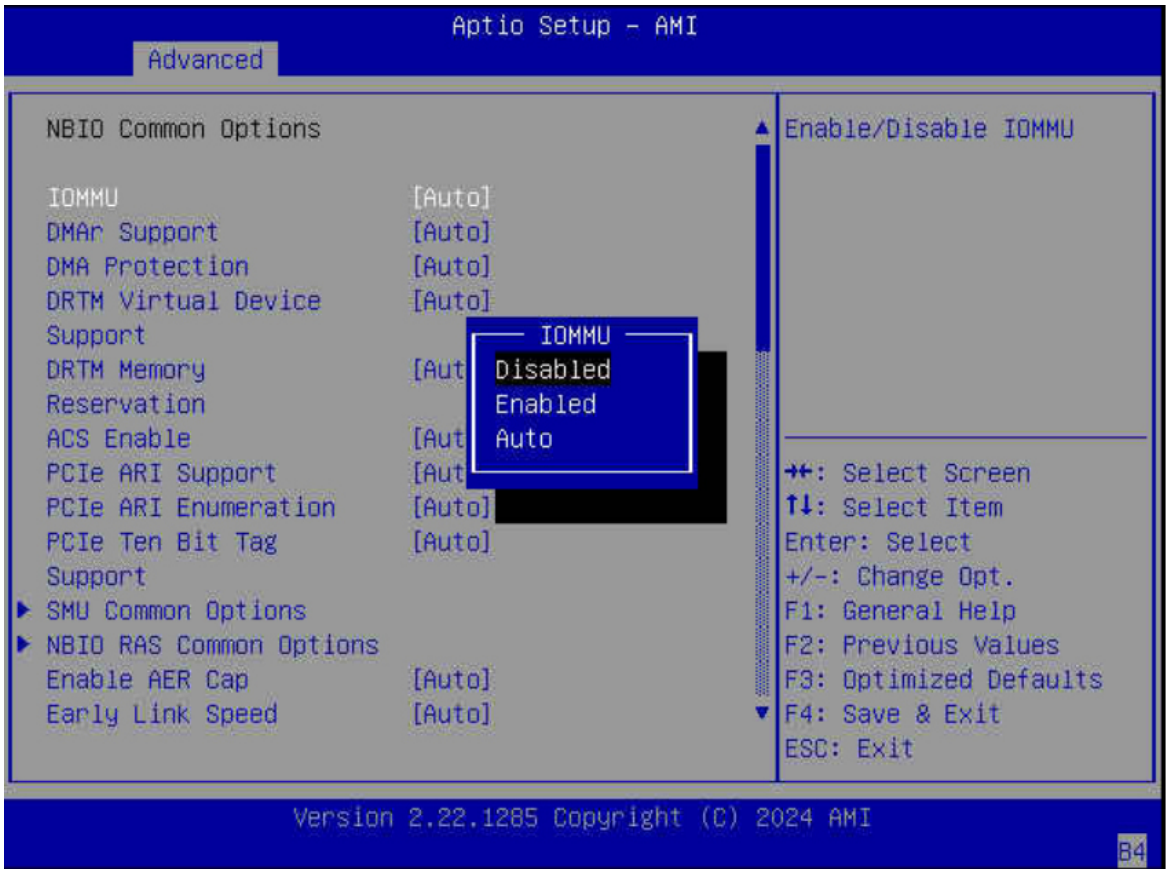
Note リブート後、iommu=pt および pci=noats パラメータがカーネル コマンド ラインに追加されます。これらのパラメータは、特定の問題に対処したり、AMD プラットフォームの互換性を向上させたりするのに役立つ場合があります。

GRUB の構成を変更すると、システムの動作に影響を与える可能性があることに注意してください。さらに問題や不安定なことが発生した場合は、変更を元に戻すか、AMD テクニカル サポートまたはディストリビューションのサポート チャネルから支援を求めることを検討してください。

方法 2： BIOS で IOMMU を無効にします。

1. BIOS で IOMMU を無効に設定：[詳細（Advance）] > [AMD CBS] > [NBIO 共通オプション（NBIO Common Options）] > [IOMMU 無効（IOMMU disable）]

Figure 7: IOMMU 無効



ROCm のアンインストール方法

解決方法:

ROCm をアンインストールするには、次の手順に従ってください。

SUMMARY STEPS

1. ターミナル ウィンドウを開きます。
2. 次のコマンドを実行して、ROCm パッケージをシステムからアンインストールします。
3. 次のコマンドを実行して、ROCm リポジトリ構成を消去します。
4. 次のコマンドを活用、amdgpu-install パッケージを削除します。
5. 最後に、次のコマンドを実行して、amd-nonfree-radeon パッケージを削除します。

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	ターミナル ウィンドウを開きます。	
ステップ 2	次のコマンドを実行して、ROCmパッケージをシステムからアンインストールします。	**$sudo amdgpu-uninstall**
ステップ 3	次のコマンドを実行して、ROCmリポジトリ構成を消去します。	**$sudo amdgpu-repo –clean**
ステップ 4	次のコマンドを活用、amdgpu-installパッケージを削除します。	**$sudo apt-get -y purge amdgpu-install**
ステップ 5	最後に、次のコマンドを実行して、amd-nonfree-radeon パッケージを削除します。	**$sudo apt-get -y purge amd-nonfree-radeon** これらの手順により、システムからROCmパッケージおよび関連コンポーネントがアンインストールされます。

What to do next

注： 提供されている特定のコマンドは、Ubuntu または Debian ベースのシステムに適していることに注意してください。別の Linux ディストリビューションを使用している場合は、パッケージマネージャ コマンドが異なる可能性があります。それに応じてコマンドを調整するか、配布で提供されている正しいパッケージ管理コマンドのマニュアルを参照してください。

これらの手順を完了したら、システムを再起動して、残りの ROCm コンポーネントが完全にクリアされていることを確認することを推奨します。

マルチ GPU システムで AMD アプリケーションが応答しなくなるのはなぜですか？

原因： 複数の GPU を備えたシステムで実行すると、GPU の使用率が 100% の状態でも、予想される GPU 温度の上昇がなくてもアプリケーションがハングします。

この問題により、アプリケーションのトランスクリプトに次のメッセージが表示されることがよくあります。

NCCL 警告カーネル コマンドラインに「iommu=pt」がないため、システムが不安定になったりハングしたりする可能性があります。

解決方法：

この問題を解決するには、/etc/default/grub の GRUB_CMDLINE_LINUX_DEFAULT に iommu=pt を追加します。

次のコマンドを実行します。

$ sudo update-grub

システムを再起動し、次のコマンドを実行します。

$ cat /proc/cmdline

返される情報には、iommu の追加を反映する必要があります。

```
BOOT_IMAGE=/vmlinuz-5.15.0-101-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro iommu=pt
```

マルチ **GPU** システムで **AMD** アプリケーションが応答しなくなるのはなぜですか？

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。