コンソールエージェントを展開する

NetApp Console setup and administration

NetApp
February 11, 2026

目次

コンソールエージェントを展開する

AWS

AWS のコンソールエージェントのインストールオプション

AWS でコンソールエージェントを作成する方法はいくつかあります。 NetApp Console から直接行うのが最も一般的な方法です。

次のインストール オプションが利用可能です。

- "コンソールから直接コンソールエージェントを作成する"（これが標準オプションです）

  このアクションにより、選択した VPC で Linux とコンソール エージェント ソフトウェアを実行する EC2 インスタンスが起動されます。

- "AWS Marketplaceからコンソールエージェントを作成する"

  このアクションでは、Linux とコンソールエージェントソフトウェアを実行する EC2 インスタンスも起動しますが、デプロイメントはコンソールからではなく AWS Marketplace から直接開始されます。

- "自分のLinuxホストにソフトウェアをダウンロードして手動でインストールする"

選択したインストール オプションは、インストールの準備方法に影響します。これには、AWS でリソースを認証および管理するために必要な権限をコンソールに付与する方法が含まれます。

NetApp ConsoleからAWSにコンソールエージェントを作成する

NetApp Consoleから直接 AWS にコンソール エージェントを作成できます。コンソールから AWS にコンソールエージェントを作成する前に、ネットワークを設定し、AWS 権限を準備する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1: AWS**にコンソールエージェントを展開するためのネットワークを設定する

コンソール エージェントをインストールする予定のネットワークの場所が次の要件をサポートしていることを確認します。これらの要件により、コンソール エージェントはハイブリッド クラウド内のリソースとプロセスを管理できるようになります。

VPCとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VPC とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
AWS サービス (amazonaws.com): • クラウドフォーメーション • エラスティックコンピューティングクラウド (EC2) • アイデンティティとアクセス管理（IAM） • キー管理サービス (KMS) • セキュリティトークンサービス（STS） • シンプルストレージサービス（S3）	AWS リソースを管理します。エンドポイントは AWS リージョンによって異なります。 "詳細についてはAWSドキュメントを参照してください"
NetApp ONTAP用の Amazon FsX: • api.workloads.netapp.com	Web ベースのコンソールは、このエンドポイントに接続して Workload Factory API と対話し、FSx for ONTAPベースのワークロードを管理および操作します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。

エンドポイント	目的
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

NetAppコンソールから接続されたエンドポイント

SaaS レイヤーを通じて提供される Web ベースのNetApp Consoleを使用すると、複数のエンドポイントに接続してデータ管理タスクが完了します。これには、コンソールからコンソール エージェントを展開するために接続されるエンドポイントが含まれます。

"NetAppコンソールから接続されたエンドポイントのリストを表示します" 。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、このネットワーク要件を実装する必要があります。

ステップ**2:** コンソールエージェントの**AWS**権限を設定する

コンソールは、VPC にコンソールエージェントをデプロイする前に、AWS で認証する必要があります。次のいずれかの認証方法を選択できます。

- コンソールに必要な権限を持つIAMロールを割り当てます
- 必要な権限を持つIAMユーザーにAWSアクセスキーとシークレットキーを提供します

どちらのオプションを使用する場合でも、最初のステップは IAM ポリシーを作成することです。このポリシーには、コンソールから AWS のコンソールエージェントを起動するために必要な権限のみが含まれています。

必要に応じて、IAMを使用してIAMポリシーを制限することができます。 `Condition`要素。 "AWS ドキュメント: 条件要素"

手順

1. AWS IAM コンソールに移動します。
2. *ポリシー > ポリシーの作成*を選択します。
3. 「JSON」を選択します。
4. 次のポリシーをコピーして貼り付けます。

   このポリシーには、コンソールから AWS のコンソールエージェントを起動するために必要な権限のみが含まれています。コンソールがコンソールエージェントを作成すると、コンソールエージェントが AWS リソースを管理できるようにする新しい権限セットがコンソールエージェントに適用されます。"コンソールエージェント自体に必要な権限の表示"。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateRole",
        "iam:DeleteRole",
        "iam:PutRolePolicy",
        "iam:CreateInstanceProfile",
        "iam:DeleteRolePolicy",
        "iam:AddRoleToInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile",
        "iam:DeleteInstanceProfile",
        "iam:PassRole",
        "iam:ListRoles",
        "ec2:DescribeInstanceStatus",
        "ec2:RunInstances",
        "ec2:ModifyInstanceAttribute",
        "ec2:CreateSecurityGroup",
        "ec2:DeleteSecurityGroup",
        "ec2:DescribeSecurityGroups",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:RevokeSecurityGroupIngress",
        "ec2:CreateNetworkInterface",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DeleteNetworkInterface",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeRegions",
        "ec2:DescribeInstances",
        "ec2:CreateTags",
        "ec2:DescribeImages",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeLaunchTemplates",
        "ec2:CreateLaunchTemplate",
        "cloudformation:CreateStack",
        "cloudformation:DeleteStack",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackEvents",
        "cloudformation:ValidateTemplate",
```

```
            "ec2:AssociateIamInstanceProfile",
            "ec2:DescribeIamInstanceProfileAssociations",
            "ec2:DisassociateIamInstanceProfile",
            "iam:GetRole",
            "iam:TagRole",
            "kms:ListAliases",
            "cloudformation:ListStacks"
        ],
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:TerminateInstances"
        ],
        "Condition": {
            "StringLike": {
                "ec2:ResourceTag/OCCMInstance": "*"
            }
        },
        "Resource": [
            "arn:aws:ec2:*:*:instance/*"
        ]
    }
  ]
}
```

5. 必要に応じて、[次へ] を選択し、タグを追加します。
6. *次へ*を選択し、名前と説明を入力します。
7. *ポリシーの作成*を選択します。
8. コンソールが引き受けることができる IAM ロールまたは IAM ユーザーにポリシーをアタッチして、コンソールにアクセスキーを提供できるようにします。
   - (オプション 1) コンソールが引き受けることができる IAM ロールを設定します。
     i. ターゲットアカウントの AWS IAM コンソールに移動します。
     ii. [アクセス管理] で、**[ロール] > [ロールの作成]** を選択し、手順に従ってロールを作成します。
     iii. *信頼されたエンティティタイプ*で、*AWS アカウント*を選択します。
     iv. *別のAWSアカウント*を選択し、コンソールSaaSアカウントのIDを入力します: 952013314444
     v. 前のセクションで作成したポリシーを選択します。
     vi. ロールを作成したら、コンソール エージェントを作成するときにコンソールに貼り付けることができるように、ロール ARN をコピーします。
   - (オプション 2) コンソールにアクセスキーを提供できるように、IAM ユーザーの権限を設定します。
     i. AWS IAM コンソールから [ユーザー] を選択し、ユーザー名を選択します。

ii. *権限の追加 > 既存のポリシーを直接添付*を選択します。

iii. 作成したポリシーを選択します。

iv. *次へ*を選択し、*権限の追加*を選択します。

v. IAM ユーザーのアクセスキーとシークレットキーがあることを確認します。

結果

これで、必要な権限を持つ IAM ロールまたは必要な権限を持つ IAM ユーザーが作成されているはずです。コンソールからコンソール エージェントを作成するときに、ロールまたはアクセス キーに関する情報を提供できます。

ステップ**3:** コンソールエージェントを作成する

コンソールの Web ベースのコンソールから直接コンソール エージェントを作成します。

タスク概要

- コンソールからコンソール エージェントを作成すると、デフォルト設定を使用して AWS に EC2 インスタンスがデプロイされます。コンソールエージェントを作成した後、CPU や RAM が少ない小さな EC2 インスタンスに切り替えないでください。"コンソールエージェントのデフォルト構成について学習します" 。
- コンソールがコンソール エージェントを作成すると、エージェントの IAM ロールとプロファイルが作成されます。このロールには、コンソールエージェントが AWS リソースを管理できるようにする権限が含まれています。将来のリリースで新しい権限が追加されたら、ロールが更新されるようにしてください。"コンソールエージェントのIAMポリシーの詳細"。

開始する前に

次のものが必要です:

- AWS 認証方法: 必要な権限を持つ IAM ユーザーの IAM ロールまたはアクセスキーのいずれか。
- ネットワーク要件を満たす VPC とサブネット。
- EC2 インスタンスのキーペア。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。
- 設定"ネットワーク要件"。
- 設定"AWS 権限"。

手順

1. *管理 > エージェント*を選択します。
2. *概要*ページで、*エージェントのデプロイ > AWS*を選択します。
3. ウィザードの手順に従ってコンソール エージェントを作成します。
4. *はじめに*ページではプロセスの概要を説明します
5. **AWS** 認証情報 ページで、AWS リージョンを指定し、認証方法 (コンソールが引き受けることができる IAM ロール、または AWS アクセスキーとシークレットキーのいずれか) を選択します。

役割を引き受ける を選択した場合は、コンソール エージェント展開ウィザードから最初の資格情報セットを作成できます。追加の資格情報セットは、「資格情報」ページから作成する必要があります。これらはウィザードのドロップダウン リストで利用できるようになります。"追加の資格情報を追加する方法を学ぶ" 。

6. *詳細*ページで、コンソール エージェントに関する詳細を入力します。
   - 名前を入力してください。
   - カスタム タグ (メタデータ) を追加します。
   - コンソールで必要な権限を持つ新しいロールを作成するか、または既存のロールを選択するかを選択します。"必要な権限" 。
   - コンソール エージェントの EBS ディスクを暗号化するかどうかを選択します。デフォルトの暗号化キーを使用するか、カスタム キーを使用するかを選択できます。
7. ネットワーク ページで、エージェントの VPC、サブネット、キーペアを指定し、パブリック IP アドレスを有効にするかどうかを選択し、必要に応じてプロキシ設定を指定します。

   コンソール エージェント仮想マシンにアクセスするための正しいキー ペアがあることを確認します。キーペアがないとアクセスできません。

8. セキュリティ グループ ページで、新しいセキュリティ グループを作成するか、必要な受信ルールと送信ルールを許可する既存のセキュリティ グループを選択するかを選択します。

   "AWS のセキュリティグループルールを表示する" 。

9. 選択内容を確認して、セットアップが正しいことを確認します。
   a. エージェント構成の検証 チェック ボックスはデフォルトでオンになっており、展開時にコンソールによってネットワーク接続要件が検証されます。コンソールがエージェントの展開に失敗した場合、トラブルシューティングに役立つレポートが提供されます。デプロイメントが成功した場合、レポートは提供されません。

   まだ使用している場合は"以前のエンドポイント"エージェントのアップグレードに使用すると、検証が失敗し、エラーが発生します。これを回避するには、チェックボックスをオフにして検証チェックをスキップします。

10. *追加*を選択します。

    コンソールは約 10 分でエージェントを展開します。プロセスが完了するまでこのページに留まります。

結果

プロセスが完了すると、コンソール エージェントはコンソールから使用できるようになります。

デプロイメントが失敗した場合は、コンソールからレポートとログをダウンロードして、問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

コンソールエージェントを作成したのと同じ AWS アカウントに Amazon S3 バケットがある場合は、システム ページに Amazon S3 作業環境が自動的に表示されます。 "NetApp Consoleから S3 バケットを管理する方

法を学びます"

AWS Marketplaceからコンソールエージェントを作成する

AWS Marketplace から直接 AWS にコンソールエージェントを作成します。 AWS Marketplace からコンソールエージェントを作成するには、ネットワークを設定し、AWS のアクセス許可を準備し、インスタンスの要件を確認してから、コンソールエージェントを作成する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ネットワークを設定する

ハイブリッド クラウド リソースを管理するには、コンソール エージェントのネットワークの場所が次の要件を満たしていることを確認します。

VPCとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VPC とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
AWS サービス (amazonaws.com): • クラウドフォーメーション • エラスティックコンピューティングクラウド (EC2) • アイデンティティとアクセス管理（IAM） • キー管理サービス (KMS) • セキュリティトークンサービス（STS） • シンプルストレージサービス（S3）	AWS リソースを管理します。エンドポイントは AWS リージョンによって異なります。 "詳細についてはAWSドキュメントを参照してください"
NetApp ONTAP用の Amazon FsX: • api.workloads.netapp.com	Web ベースのコンソールは、このエンドポイントに接続して Workload Factory API と対話し、FSx for ONTAPベースのワークロードを管理および操作します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。

エンドポイント	目的
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォール ルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インター

ネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、このネットワーク アクセスを実装します。

ステップ**2: AWS**権限を設定する

マーケットプレイスの展開を準備するには、AWS で IAM ポリシーを作成し、それを IAM ロールにアタッチします。 AWS Marketplace からコンソールエージェントを作成すると、その IAM ロールを選択するように求められます。

手順

1. AWS コンソールにログインし、IAM サービスに移動します。
2. ポリシーを作成します。
   a. *ポリシー > ポリシーの作成*を選択します。
   b. *JSON*を選択し、その内容をコピーして貼り付けます。"コンソールエージェントのIAMポリシー" 。
   c. 残りの手順を完了してポリシーを作成します。

      使用する予定のNetAppデータ サービスに基づいて、2 番目のポリシーを作成する必要がある場合があります。標準リージョンの場合、権限は 2 つのポリシーに分散されます。 AWS の管理ポリシーの最大文字サイズ制限により、2 つのポリシーが必要になります。"コンソールエージェントのIAMポリシーの詳細" 。
3. IAM ロールを作成します。
   a. *[ロール] > [ロールの作成]*を選択します。
   b. **AWS** サービス **> EC2** を選択します。
   c. 作成したポリシーを添付して権限を追加します。
   d. 残りの手順を完了してロールを作成します。

結果

これで、AWS Marketplace からのデプロイ時に EC2 インスタンスに関連付けることができる IAM ロールが作成されました。

ステップ**3:** インスタンス要件を確認する

コンソールエージェントを作成するときは、次の要件を満たす EC2 インスタンスタイプを選択する必要があります。

CPU

8コアまたは8vCPU

RAM

32 GB

AWS EC2インスタンスタイプ

CPU と RAM の要件を満たすインスタンス タイプ。 NetAppt3.2xlarge を推奨します。

ステップ**4:** コンソールエージェントを作成する

AWS Marketplace から直接コンソールエージェントを作成します。

タスク概要

AWS Marketplace からコンソールエージェントを作成すると、デフォルト設定を使用して AWS に EC2 インスタンスがデプロイされます。"コンソールエージェントのデフォルト構成について学習します" 。

開始する前に

次のものが必要です:

- ネットワーク要件を満たす VPC とサブネット。
- コンソール エージェントに必要な権限を含むポリシーがアタッチされた IAM ロール。
- IAM ユーザーが AWS Marketplace にサブスクライブしたり、サブスクライブ解除したりするための権限。
- インスタンスの CPU および RAM 要件を理解すること。
- EC2 インスタンスのキーペア。

手順

1. に行く "AWS Marketplace でのNetApp Consoleエージェントのリスト"
2. マーケットプレイス ページで、[サブスクリプションを続行] を選択します。
3. ソフトウェアをサブスクライブするには、「利用規約に同意」を選択します。

   サブスクリプションのプロセスには数分かかる場合があります。

4. サブスクリプションプロセスが完了したら、[構成に進む] を選択します。
5. *このソフトウェアを構成する*ページで、正しいリージョンが選択されていることを確認し、*起動を続行*を選択します。
6. このソフトウェアの起動 ページの アクションの選択 で、**EC2** 経由で起動 を選択し、起動 を選択します。

   EC2 コンソールを使用してインスタンスを起動し、IAM ロールをアタッチします。これは、**Web** サイトから起動 アクションでは不可能です。

7. プロンプトに従ってインスタンスを構成してデプロイします。

   ◦ 名前とタグ: インスタンスの名前とタグを入力します。

- アプリケーションと **OS** イメージ: このセクションはスキップします。コンソール エージェント AMI はすでに選択されています。
- インスタンス タイプ: リージョンの可用性に応じて、RAM と CPU の要件を満たすインスタンス タイプを選択します (t3.2xlarge が事前に選択されており、推奨されています)。
- キーペア **(**ログイン**)**: インスタンスに安全に接続するために使用するキーペアを選択します。
- ネットワーク設定: 必要に応じてネットワーク設定を編集します。
  - 必要な VPC とサブネットを選択します。
  - インスタンスにパブリック IP アドレスを割り当てるかどうかを指定します。
  - コンソール エージェント インスタンスに必要な接続方法 (SSH、HTTP、HTTPS) を有効にするセキュリティ グループ設定を指定します。

    "AWS のセキュリティグループルールを表示する" 。
- ストレージの構成: ルート ボリュームのデフォルトのサイズとディスク タイプを維持します。

  ルートボリュームで Amazon EBS 暗号化を有効にする場合は、[詳細] を選択し、[ボリューム **1**] を展開して、[暗号化] を選択し、KMS キーを選択します。
- 詳細: **IAM** インスタンス プロファイル で、コンソール エージェントに必要な権限を含む IAM ロールを選択します。
- 概要: 概要を確認し、*インスタンスの起動*を選択します。

  AWS は指定された設定でコンソールエージェントを起動し、コンソールエージェントは約 10 分で実行されます。

> インストールが失敗した場合は、トラブルシューティングに役立つログとレポートを表示できます。"インストールの問題をトラブルシューティングする方法を学びます。"

8. コンソール エージェント仮想マシンに接続しているホストとコンソール エージェントの URL から Web ブラウザーを開きます。
9. ログイン後、コンソール エージェントを設定します。
   a. コンソール エージェントに関連付けるコンソール組織を指定します。
   b. システムの名前を入力します。
   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      コンソールを標準モードで使用するには、制限モードを無効にしておきます。安全な環境があり、このアカウントをコンソールのバックエンド サービスから切断する場合にのみ、制限モードを有効にする必要があります。もしそうなら、"NetApp Consoleを制限モードで使い始めるための手順" 。
   d. *始めましょう*を選択します。

結果

コンソール エージェントがインストールされ、コンソール組織に設定されました。

ウェブブラウザを開いて、 "NetApp Console"コンソールでコンソール エージェントの使用を開始します。

コンソールエージェントを作成したのと同じ AWS アカウントに Amazon S3 バケットがある場合は、システム ページに Amazon S3 作業環境が自動的に表示されます。 "NetApp Consoleから S3 バケットを管理する方法を学びます"

AWSにコンソールエージェントを手動でインストールする

AWS で実行されている Linux ホストにコンソールエージェントを手動でインストールできます。独自の Linux ホストにコンソールエージェントを手動でインストールするには、ホスト要件を確認し、ネットワークを設定し、AWS 権限を準備し、コンソールエージェントをインストールして、準備した権限を付与する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ホストの要件を確認する

コンソール エージェント ソフトウェアを実行しているホストが、オペレーティング システム、RAM、およびポートの要件を満たしていることを確認します。

> コンソール エージェントは、19000 ～ 19200 の UID と GID の範囲を予約します。この範囲は固定されており、変更することはできません。ホスト上のサードパーティ ソフトウェアがこの範囲内の UID または GID を使用している場合、エージェントのインストールは失敗します。NetApp、競合を回避するためにサードパーティ ソフトウェアがインストールされていないホストの使用を推奨しています。

専用ホスト

コンソール エージェントには専用のホストが必要です。次のサイズ要件を満たすアーキテクチャであれば、どれでもサポートされます。

- CPU: 8コアまたは8vCPU
- メモリ: 32 GB
- ディスク容量: ホストには 165 GB が推奨され、パーティション要件は次のとおりです。
  - `/opt`: 120 GiBの空き容量が必要です

    エージェントは `/opt`インストールするには `/opt/application/netapp`ディレクトリとその内容。
  - `/var`: 40 GiBの空き容量が必要です

    コンソールエージェントにはこのスペースが必要です `/var`Podman または Docker は、このディレクトリ内にコンテナを作成するように設計されているためです。具体的には、`/var/lib/containers/storage`ディレクトリと `/var/lib/docker`Docker用。このスペースでは外部マウントまたはシンボリックリンクは機能しません。

AWS EC2インスタンスタイプ

CPU と RAM の要件を満たすインスタンス タイプ。 NetAppt3.2xlarge を推奨します。

ハイパーバイザー

サポートされているオペレーティング システムを実行することが認定されているベア メタルまたはホスト型ハイパーバイザーが必要です。

オペレーティングシステムとコンテナの要件

コンソールを標準モードまたは制限モードで使用する場合、コンソール エージェントは次のオペレーティング システムでサポートされます。エージェントをインストールする前に、コンテナ オーケストレーション ツールが必要です。

オペレーティング システム	サポートされる**OS** バージョン	サポートされているエージェントのバージョン	必要なコンテナツール	**SELinux**
Red Hat Enterprise Linux		9.6 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	4.0.0 以降、コンソールが標準モードまたは制限モード	Podman バージョン 5.4.0 と podman-compose 1.5.0。 Podman の構成要件を表示する 。

オペレーティングシステム	サポートされるOSバージョン	サポートされているエージェントのバージョン	必要なコンテナツール	SELinux
強制モードまたは許可モードでサポートされます		9.1～9.4 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.9.4 と podman-compose 1.5.0。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます		8.6～8.10 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.6.1 または 4.9.4 と podman-compose 1.0.6。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます	Ubuntu		24.04 LTS	3.9.45 以降、NetApp Consoleが標準モードまたは制限モード
Docker エンジン 23.06 から 28.0.0。	サポート対象外		22.04 LTS	3.9.50以降

鍵ペア

コンソールエージェントを作成するときは、インスタンスで使用する EC2 キーペアを選択する必要があります。

IMDSv2 使用時の **PUT** 応答ホップ制限

IMDSv2 が有効になっている場合 (新しい EC2 インスタンスのデフォルト)、PUT 応答ホップ制限を 3 に設定します。そうしないと、エージェントのセットアップ中に UI 初期化エラーが表示されます。

- "Amazon EC2 インスタンスで IMDSv2 の使用を必須にする"
- "AWSドキュメント: PUTレスポンスのホップ制限を変更する"

ステップ**2: Podman**または**Docker Engine**をインストールする

オペレーティング システムに応じて、エージェントをインストールする前に Podman または Docker Engine のいずれかが必要になります。

- Red Hat Enterprise Linux 8 および 9 には Podman が必要です。

  サポートされているPodmanのバージョンを表示する 。

- Ubuntu には Docker Engine が必要です。

  サポートされている Docker エンジンのバージョンを表示する 。

例 **1.** 手順

ポッドマン

Podman をインストールして設定するには、次の手順に従います。

- podman.socket サービスを有効にして起動します
- Python3をインストールする
- podman-compose パッケージ バージョン 1.0.6 をインストールします。
- podman-composeをPATH環境変数に追加する
- Red Hat Enterprise Linux を使用している場合は、Podman バージョンが CNI ではなく Netavark Aardvark DNS を使用していることを確認してください。

DNS ポートの競合を避けるために、エージェントをインストールした後、aardvark-dns ポート (デフォルト: 53) を調整します。指示に従ってポートを構成します。

手順

1. ホストに podman-docker パッケージがインストールされている場合は削除します。

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Podman をインストールします。

   Podman は、公式の Red Hat Enterprise Linux リポジトリから入手できます。

   a. Red Hat Enterprise Linux 9.6 の場合:

   ```
   sudo dnf install podman-5:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   b. Red Hat Enterprise Linux 9.1 から 9.4 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   c. Red Hat Enterprise Linux 8 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

<version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

3. podman.socket サービスを有効にして起動します。

```
sudo systemctl enable --now podman.socket
```

4. python3 をインストールします。

```
sudo dnf install python3
```

5. システムにまだインストールされていない場合は、EPEL リポジトリ パッケージをインストールします。

   podman-compose は、Extra Packages for Enterprise Linux (EPEL) リポジトリから入手できるため、この手順は必須です。

6. Red Hat Enterprise 9 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

   +

   a. podman-compose パッケージ 1.5.0 をインストールします。

```
sudo dnf install podman-compose-1.5.0
```

7. Red Hat Enterprise Linux 8 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

   b. podman-compose パッケージ 1.0.6 をインストールします。

```
sudo dnf install podman-compose-1.0.6
```

使用して `dnf install`コマンドは、PATH 環境変数に podman-compose を追加するための要件を満たしています。インストールコマンドは、すでにインストールされているpodman-composeを/usr/binに追加します。 `secure_path`ホスト上のオプション。

c. Red Hat Enterprise Linux 8 を使用している場合は、Podman バージョンが CNI ではなく Aardvark DNS を備えた NetAvark を使用していることを確認します。

i. 次のコマンドを実行して、networkBackend が CNI に設定されているかどうかを確認します。

```
podman info | grep networkBackend
```

ii. networkBackendが `CNI`、それを変更する必要があります `netavark`。

iii. インストール `netavark`そして `aardvark-dns`次のコマンドを使用します。

```
dnf install aardvark-dns netavark
```

iv. 開く `/etc/containers/containers.conf`ファイルを編集し、network_backend オプションを変更して、「cni」の代わりに「netavark」を使用します。

もし `/etc/containers/containers.conf`存在しない場合は、設定を変更してください `/usr/share/containers/containers.conf`。

v. podman を再起動します。

```
systemctl restart podman
```

vi. 次のコマンドを使用して、networkBackend が「netavark」に変更されていることを確認します。

```
podman info | grep networkBackend
```

Dockerエンジン

Docker のドキュメントに従って Docker Engine をインストールします。

手順

1. "Dockerからのインストール手順を見る"

   サポートされている Docker エンジン バージョンをインストールするには、手順に従ってください。最新バージョンはコンソールでサポートされていないため、インストールしないでください。

2. Docker が有効になっていて実行されていることを確認します。

```
sudo systemctl enable docker && sudo systemctl start docker
```

ステップ**3:** ネットワークを設定する

コンソール エージェントがハイブリッド クラウド内のリソースを管理できるように、ネットワークの場所が次の要件をサポートしていることを確認します。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

Webベースの**NetApp Console**を使用する際にコンピュータから接続されるエンドポイント

Web ブラウザからコンソールにアクセスするコンピュータは、複数のエンドポイントに接続できる必要があります。コンソール エージェントを設定し、コンソールを日常的に使用するには、コンソールを使用する必要があります。

"NetAppコンソールのネットワークを準備する" 。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
AWS サービス (amazonaws.com): • クラウドフォーメーション • エラスティックコンピューティングクラウド (EC2) • アイデンティティとアクセス管理（IAM） • キー管理サービス (KMS) • セキュリティトークンサービス（STS） • シンプルストレージサービス（S3）	AWS リソースを管理します。エンドポイントは AWS リージョンによって異なります。 "詳細についてはAWSドキュメントを参照してください"

エンドポイント	目的
NetApp ONTAP用の Amazon FsX: • api.workloads.netapp.com	Web ベースのコンソールは、このエンドポイントに接続して Workload Factory API と対話し、FSx for ONTAPベースのワークロードを管理および操作します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。

エンドポイント	目的
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォール ルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インター

ネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

ステップ**4:** コンソールの**AWS**権限を設定する

次のいずれかのオプションを使用して、 NetApp Consoleに AWS 権限を付与します。

- オプション 1: IAM ポリシーを作成し、EC2 インスタンスに関連付けることができる IAM ロールにポリシーをアタッチします。
- オプション 2: 必要な権限を持つ IAM ユーザーの AWS アクセスキーをコンソールに提供します。

コンソールの権限を準備するには、手順に従ってください。

IAMのロール

手順

1. AWS コンソールにログインし、IAM サービスに移動します。
2. ポリシーを作成します。
   a. *ポリシー > ポリシーの作成*を選択します。
   b. *JSON*を選択し、その内容をコピーして貼り付けます。"コンソールエージェントのIAMポリシー" 。
   c. 残りの手順を完了してポリシーを作成します。

      使用する予定のNetAppデータ サービスによっては、2 番目のポリシーを作成する必要がある場合があります。標準リージョンの場合、権限は 2 つのポリシーに分散されます。 AWS の管理ポリシーの最大文字サイズ制限により、2 つのポリシーが必要になります。"コンソールエージェントのIAMポリシーの詳細" 。
3. IAM ロールを作成します。
   a. *[ロール] > [ロールの作成]*を選択します。
   b. **AWS** サービス **> EC2** を選択します。
   c. 作成したポリシーを添付して権限を追加します。
   d. 残りの手順を完了してロールを作成します。

結果

コンソールエージェントをインストールした後、EC2 インスタンスに関連付けることができる IAM ロールが作成されます。

AWS アクセスキー

手順

1. AWS コンソールにログインし、IAM サービスに移動します。
2. ポリシーを作成します。
   a. *ポリシー > ポリシーの作成*を選択します。
   b. *JSON*を選択し、その内容をコピーして貼り付けます。"コンソールエージェントのIAMポリシー" 。
   c. 残りの手順を完了してポリシーを作成します。

      使用する予定のNetAppデータ サービスによっては、2 番目のポリシーを作成する必要がある場合があります。

   標準リージョンの場合、権限は 2 つのポリシーに分散されます。 AWS の管理ポリシーの最大文字サイズ制限により、2 つのポリシーが必要になります。"コンソールエージェントのIAMポリシーの詳細" 。
3. IAM ユーザーにポリシーをアタッチします。
   - "AWSドキュメント: IAMロールの作成"

- "AWSドキュメント: IAMポリシーの追加と削除"

4. コンソール エージェントをインストールした後、 NetApp Consoleに追加できるアクセス キーがユーザーにあることを確認します。

結果

これで、必要な権限を持つ IAM ユーザーと、コンソールに提供できるアクセス キーが作成されました。

ステップ**5:** コンソールエージェントをインストールする

前提条件を満たしたら、Linux ホストにソフトウェアを手動でインストールします。

開始する前に

次のものが必要です:

- コンソール エージェントをインストールするためのルート権限。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。

  インストール後にプロキシ サーバーを構成するオプションがありますが、これを行うにはコンソール エージェントを再起動する必要があります。

- プロキシ サーバーが HTTPS を使用する場合、またはプロキシがインターセプト プロキシである場合は、CA 署名証明書。

> コンソール エージェントを手動でインストールする場合、透過プロキシ サーバーの証明書を設定することはできません。透過プロキシ サーバーの証明書を設定する必要がある場合は、インストール後にメンテナンス コンソールを使用する必要があります。詳細はこちら "エージェントメンテナンスコンソール"。

タスク概要

インストール後、新しいバージョンが利用可能な場合、コンソール エージェントは自動的に更新されます。

手順

1. ホストに *http_proxy* または *https_proxy* システム変数が設定されている場合は、それらを削除します。

```
unset http_proxy
unset https_proxy
```

   これらのシステム変数を削除しないと、インストールは失敗します。

2. コンソール エージェント ソフトウェアをダウンロードし、Linux ホストにコピーします。 NetApp ConsoleまたはNetAppサポート サイトからダウンロードできます。
   - NetApp Console: エージェント **>** 管理 **>** エージェントのデプロイ **>** オンプレミス **>** 手動インストールに移動します。

     エージェント インストーラー ファイルのダウンロードまたはファイルへの URL を選択します。

◦ NetAppサポート サイト (コンソールにまだアクセスできない場合に必要) "NetAppサポート サイト" 、

3. スクリプトを実行するための権限を割り当てます。

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

<version> は、ダウンロードしたコンソール エージェントのバージョンです。

4. Government Cloud 環境にインストールする場合は、構成チェックを無効にします。"手動インストールの構成チェックを無効にする方法を説明します。"
5. インストール スクリプトを実行します。

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy
server> --cacert <path and file name of a CA-signed certificate>
```

ネットワークでインターネットアクセスにプロキシが必要な場合は、プロキシ情報を追加する必要があります。インストール中に明示的にプロキシを追加できます。 `--proxy`および `--cacert`パラメータはオプションであり、追加を要求されることはありません。明示的なプロキシ サーバがある場合は、示されているようにパラメータを入力する必要があります。

透過プロキシを設定する場合は、インストール後に設定できます。 "エージェントメンテナンスコンソールについて学ぶ"

+

CA 署名証明書を使用して明示的なプロキシ サーバーを構成する例を次に示します。

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+
`--proxy` 次のいずれかの形式を使用して、Console エージェントが HTTP または HTTPS プロキシ サーバを使用するように設定します：

+ * http://address:port * http://user-name:password@address:port * http://domain-name%92user-name:password@address:port * https://address:port * https://user-name:password@address:port * https://domain-name%92user-name:password@address:port

+ 以下の点に注意してください：

+ ユーザーは、ローカル ユーザーまたはドメイン ユーザーにすることができます。 ドメイン ユーザーの場合は、上記のように \ の ASCII コードを使用する必要があります。 **Console** エージェントは、**@** 文字を含むユーザー名またはパスワードをサポートしていません。 パスワードに次の特殊文字が含まれている場合は、その特殊文字の前にバックスラッシュを付けてエスケープする必要があります：& または !

+ 例：

+ http://bxpproxyuser:netapp1\!@address:3128

1. Podman を使用した場合は、aardvark-dns ポートを調整する必要があります。

   a. コンソール エージェント仮想マシンに SSH で接続します。

   b. podman */usr/share/containers/containers.conf* ファイルを開き、Aardvark DNS サービス用に選択したポートを変更します。たとえば、54 に変更します。

      ```
      vi /usr/share/containers/containers.conf
      ```

      例えば：

      ```
      # Port to use for dns forwarding daemon with netavark in rootful bridge
      # mode and dns enabled.
      # Using an alternate port might be useful if other DNS services should
      # run on the machine.
      #
      dns_bind_port = 54
      ```

   a. コンソール エージェント仮想マシンを再起動します。

2. インストールが完了するまでお待ちください。

   プロキシ サーバーを指定した場合、インストールの最後に、コンソール エージェント サービス (occm) が 2 回再起動します。

   インストールが失敗した場合は、インストール レポートとログを表示して問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

1. コンソール エージェント仮想マシンに接続しているホストから Web ブラウザを開き、次の URL を入力します。

   https://ipaddress

2. ログイン後、コンソール エージェントを設定します。

   a. コンソール エージェントに関連付ける組織を指定します。

   b. システムの名前を入力します。

   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      以下の手順ではコンソールを標準モードで使用する方法について説明しているため、制限モードは無効にしておく必要があります。安全な環境があり、このアカウントをバックエンド サービスから切断する場合にのみ、制限モードを有効にする必要があります。もしそうなら、"NetApp Consoleを制限モードで使い始めるための手順に従います" 。

d. *始めましょう*を選択します。

コンソールエージェントを作成したのと同じ AWS アカウントに Amazon S3 バケットがある場合は、[システム] ページに Amazon S3 ストレージ システムが自動的に表示されます。 "NetApp ConsolePからS3バケットを管理する方法を学ぶ"

ステップ**6: NetApp Console**に権限を付与する

コンソールエージェントをインストールした後、コンソールエージェントが AWS 内のデータとストレージインフラストラクチャを管理できるように、設定した AWS 権限を提供します。

IAMのロール

作成した IAM ロールをコンソールエージェント EC2 インスタンスにアタッチします。

手順

1. Amazon EC2 コンソールに移動します。
2. *インスタンス*を選択します。
3. コンソール エージェント インスタンスを選択します。
4. *アクション > セキュリティ > IAM ロールの変更*を選択します。
5. IAM ロールを選択し、*IAM ロールの更新*を選択します。

に行く "NetApp Console"コンソール エージェントの使用を開始します。

AWS アクセスキー

必要な権限を持つ IAM ユーザーの AWS アクセスキーをコンソールに提供します。

手順

1. コンソールで正しいコンソール エージェントが現在選択されていることを確認します。
2. *管理 > 資格情報*を選択します。
3. *組織の資格情報*を選択します。
4. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: *Amazon Web Services > エージェントを選択します。
   b. 資格情報の定義: AWS アクセスキーとシークレットキーを入力します。
   c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。
   d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

に行く "NetApp Console"コンソール エージェントの使用を開始します。

Azure

Azure のコンソール エージェントのインストール オプション

Azure でコンソール エージェントを作成するには、いくつかの方法があります。NetApp Consoleから直接行うのが最も一般的な方法です。

次のインストール オプションが利用可能です。

- "NetApp Consoleから直接コンソールエージェントを作成する"（これが標準オプションです）

  このアクションにより、選択した VNet で Linux とコンソール エージェント ソフトウェアを実行する VM が起動します。

- "Azure Marketplace からコンソール エージェントを作成する"

  このアクションでは、Linux とコンソール エージェント ソフトウェアを実行する VM も起動しますが、展開はコンソールからではなく、Azure Marketplace から直接開始されます。

- "自分のLinuxホストにソフトウェアをダウンロードして手動でインストールする"

選択したインストール オプションは、インストールの準備方法に影響します。これには、Azure でリソースを認証および管理するために必要なアクセス許可をコンソール エージェントに付与する方法が含まれます。

NetApp Consoleから Azure にコンソール エージェントを作成する

NetApp Consoleから Azure にコンソール エージェントを作成するには、ネットワークを設定し、Azure 権限を準備してから、コンソール エージェントを作成する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ネットワークを設定する

コンソール エージェントをインストールする予定のネットワークの場所が次の要件をサポートしていることを確認します。これらの要件により、コンソール エージェントはハイブリッド クラウド リソースを管理できるようになります。

Azureリージョン

Cloud Volumes ONTAPを使用する場合、コンソールエージェントは、管理するCloud Volumes ONTAPシステムと同じAzureリージョン、または "Azure リージョン ペア"Cloud Volumes ONTAPシステム用。この要件により、 Cloud Volumes ONTAPとそれに関連付けられたストレージ アカウント間で Azure Private Link 接続が使用されるようになります。

"Cloud Volumes ONTAP がAzure Private Link を使用する方法を学ぶ"

VNetとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VNet とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	Azure パブリック リージョン内のリソースを管理します。
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	Azure China リージョンのリソースを管理します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェア アップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。

エンドポイント	目的
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォール ルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

NetAppコンソールから接続されたエンドポイント

SaaS レイヤーを通じて提供される Web ベースのNetApp Consoleを使用すると、複数のエンドポイントに接続してデータ管理タスクが完了します。これには、コンソールからコンソール エージェントを展開するために接続されるエンドポイントが含まれます。

"NetAppコンソールから接続されたエンドポイントのリストを表示します" 。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、このネットワーク要件を実装する必要があります。

ステップ **2:** コンソール エージェント展開ポリシー **(**カスタム ロール**)** を作成する

Azure にコンソール エージェントをデプロイする権限を持つカスタム ロールを作成する必要があります。

Azure アカウントまたは Microsoft Entra サービス プリンシパルに割り当てることができる Azure カスタム ロールを作成します。コンソールは Azure で認証し、これらのアクセス許可を使用してユーザーに代わってコンソール エージェントを作成します。

コンソールはAzureにコンソールエージェントVMを展開し、 "システム割り当てマネージドID"必要なロールを作成し、それを VM に割り当てます。"コンソールが権限をどのように使用するかを確認します" 。

Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

手順

1. Azure の新しいカスタム ロールに必要なアクセス許可をコピーし、JSON ファイルに保存します。

   このカスタム ロールには、コンソールから Azure のコンソール エージェント VM を起動するために必要なアクセス許可のみが含まれています。このポリシーを他の状況では使用しないでください。コンソールは、コンソール エージェントを作成するときに、コンソール エージェントが Azure リソースを管理できるようにする新しいアクセス許可セットをコンソール エージェント VM に適用します。

```
{
  "Name": "Azure SetupAsService",
  "Actions": [
    "Microsoft.Compute/disks/delete",
    "Microsoft.Compute/disks/read",
    "Microsoft.Compute/disks/write",
    "Microsoft.Compute/locations/operations/read",
```

```
    "Microsoft.Compute/operations/read",
    "Microsoft.Compute/virtualMachines/instanceView/read",
    "Microsoft.Compute/virtualMachines/read",
    "Microsoft.Compute/virtualMachines/write",
    "Microsoft.Compute/virtualMachines/delete",
    "Microsoft.Compute/virtualMachines/extensions/write",
    "Microsoft.Compute/virtualMachines/extensions/read",
    "Microsoft.Compute/availabilitySets/read",
    "Microsoft.Network/locations/operationResults/read",
    "Microsoft.Network/locations/operations/read",
    "Microsoft.Network/networkInterfaces/join/action",
    "Microsoft.Network/networkInterfaces/read",
    "Microsoft.Network/networkInterfaces/write",
    "Microsoft.Network/networkInterfaces/delete",
    "Microsoft.Network/networkSecurityGroups/join/action",
    "Microsoft.Network/networkSecurityGroups/read",
    "Microsoft.Network/networkSecurityGroups/write",
    "Microsoft.Network/virtualNetworks/checkIpAddressAvailability/read",
    "Microsoft.Network/virtualNetworks/read",
    "Microsoft.Network/virtualNetworks/subnets/join/action",
    "Microsoft.Network/virtualNetworks/subnets/read",
    "Microsoft.Network/virtualNetworks/subnets/virtualMachines/read",
    "Microsoft.Network/virtualNetworks/virtualMachines/read",
    "Microsoft.Network/publicIPAddresses/write",
    "Microsoft.Network/publicIPAddresses/read",
    "Microsoft.Network/publicIPAddresses/delete",
    "Microsoft.Network/networkSecurityGroups/securityRules/read",
    "Microsoft.Network/networkSecurityGroups/securityRules/write",
    "Microsoft.Network/networkSecurityGroups/securityRules/delete",
    "Microsoft.Network/publicIPAddresses/join/action",

"Microsoft.Network/locations/virtualNetworkAvailableEndpointServices/rea
d",
    "Microsoft.Network/networkInterfaces/ipConfigurations/read",
    "Microsoft.Resources/deployments/operations/read",
    "Microsoft.Resources/deployments/read",
    "Microsoft.Resources/deployments/delete",
    "Microsoft.Resources/deployments/cancel/action",
    "Microsoft.Resources/deployments/validate/action",
    "Microsoft.Resources/resources/read",
    "Microsoft.Resources/subscriptions/operationresults/read",
    "Microsoft.Resources/subscriptions/resourceGroups/delete",
    "Microsoft.Resources/subscriptions/resourceGroups/read",
    "Microsoft.Resources/subscriptions/resourcegroups/resources/read",
    "Microsoft.Resources/subscriptions/resourceGroups/write",
    "Microsoft.Authorization/roleDefinitions/write",
```

```
    "Microsoft.Authorization/roleAssignments/write",

"Microsoft.MarketplaceOrdering/offertypes/publishers/offers/plans/agreements/read",

"Microsoft.MarketplaceOrdering/offertypes/publishers/offers/plans/agreements/write",
    "Microsoft.Network/networkSecurityGroups/delete",
    "Microsoft.Storage/storageAccounts/delete",
    "Microsoft.Storage/storageAccounts/write",
    "Microsoft.Resources/deployments/write",
    "Microsoft.Resources/deployments/operationStatuses/read",
    "Microsoft.Authorization/roleAssignments/read"
  ],
  "NotActions": [],
  "AssignableScopes": [],
  "Description": "Azure SetupAsService",
  "IsCustom": "true"
}
```

2. 割り当て可能なスコープに Azure サブスクリプション ID を追加して JSON を変更します。

   例

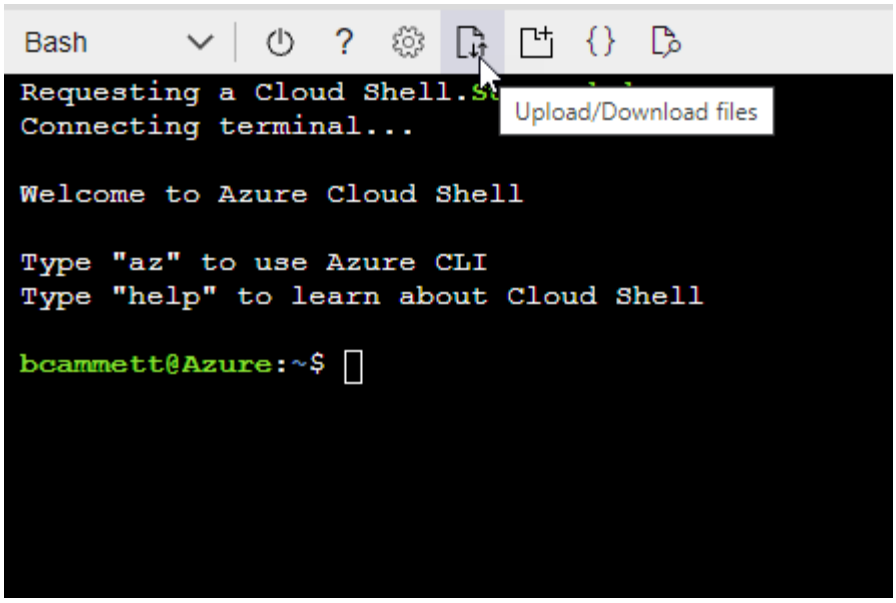
```
"AssignableScopes": [
"/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz"
]
```

3. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

   次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

   a. 始める "Azure クラウド シェル"Bash 環境を選択します。

   b. JSON ファイルをアップロードします。



c. 次の Azure CLI コマンドを入力します。

```
az role definition create --role-definition
Policy_for_Setup_As_Service_Azure.json
```

これで、*Azure SetupAsService* というカスタム ロールが作成されました。このカスタム ロールは、ユーザー アカウントまたはサービス プリンシパルに適用できます。

ステップ**3:** 認証を設定する

コンソールからコンソール エージェントを作成するときは、コンソールが Azure で認証して VM をデプロイできるようにするためのログインを提供する必要があります。次の 2 つのオプションがあります。

1. プロンプトが表示されたら、Azure アカウントでSign in。このアカウントには特定の Azure 権限が必要です。これがデフォルトのオプションです。
2. Microsoft Entra サービス プリンシパルに関する詳細を提供します。このサービス プリンシパルには特定のアクセス許可も必要です。

コンソールで使用するために、これらの認証方法のいずれかを準備するには、手順に従ってください。

Azureアカウント

コンソールからコンソール エージェントを展開するユーザーにカスタム ロールを割り当てます。

手順

1. Azure ポータルで、サブスクリプション サービスを開き、ユーザーのサブスクリプションを選択します。
2. アクセス制御 **(IAM)** をクリックします。
3. 追加 > ロール割り当ての追加 をクリックし、権限を追加します。
   a. **Azure SetupAsService** ロールを選択し、次へ をクリックします。

   Azure SetupAsService は、Azure のコンソール エージェント展開ポリシーで提供される既定の名前です。ロールに別の名前を選択した場合は、代わりにその名前を選択します。

   b. *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
   c. *メンバーを選択*をクリックし、ユーザーアカウントを選択して*選択*をクリックします。
   d. *次へ*をクリックします。
   e. *レビュー+割り当て*をクリックします。

サービスプリンシパル

Azure アカウントでログインするのではなく、必要な権限を持つ Azure サービス プリンシパルの資格情報をコンソールに提供できます。

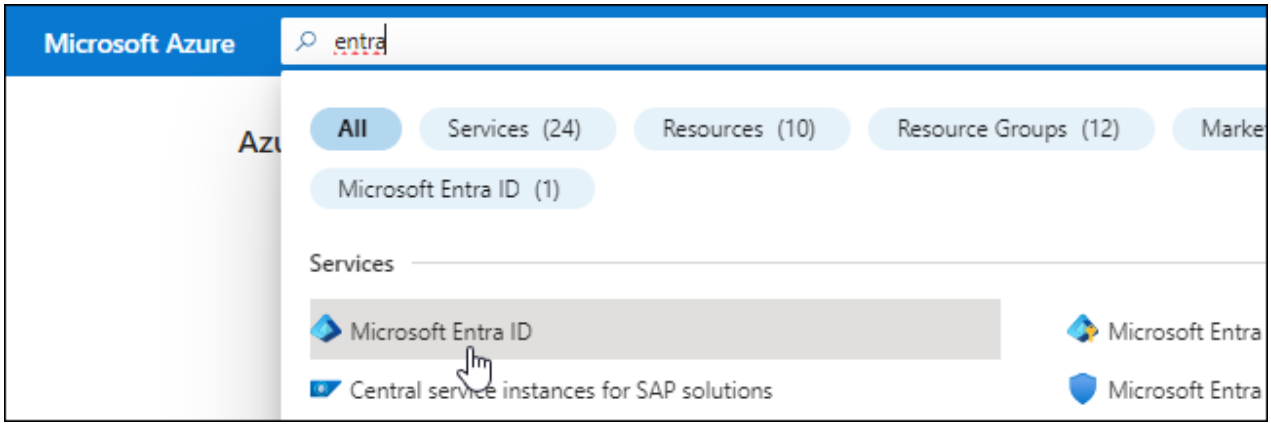
Microsoft Entra ID でサービス プリンシパルを作成して設定し、コンソールに必要な Azure 資格情報を取得します。

ロールベースのアクセス制御用の **Microsoft Entra** アプリケーションを作成する

1. Azure で Active Directory アプリケーションを作成し、そのアプリケーションをロールに割り当てるためのアクセス許可があることを確認します。

   詳細については、 "Microsoft Azure ドキュメント: 必要な権限"

2. Azure ポータルから、**Microsoft Entra ID** サービスを開きます。

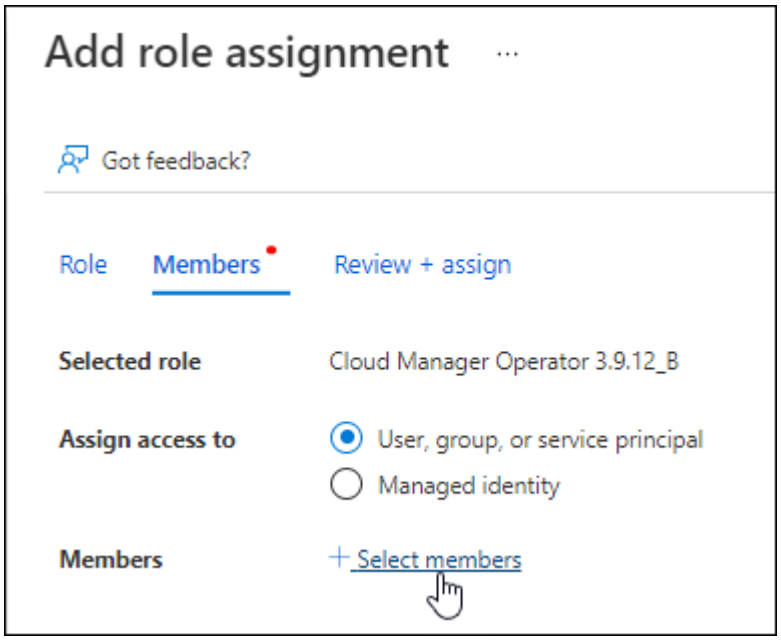


3. メニューで*アプリ登録*を選択します。
4. *新規登録*を選択します。
5. アプリケーションの詳細を指定します。
   - 名前: アプリケーションの名前を入力します。
   - アカウント タイプ: アカウント タイプを選択します (いずれのタイプもNetApp Consoleで使用できます)。
   - リダイレクト **URI**: このフィールドは空白のままにすることができます。
6. *登録*を選択します。

   AD アプリケーションとサービス プリンシパルを作成しました。

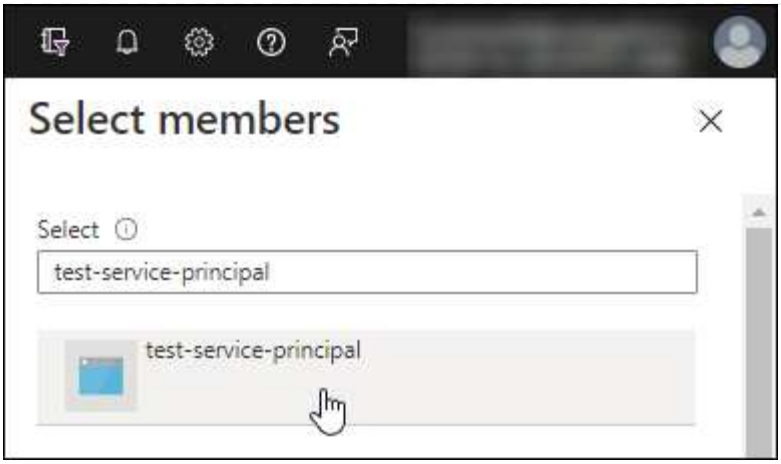
アプリケーションにカスタムロールを割り当てる

1. Azure ポータルから、サブスクリプション サービスを開きます。
2. サブスクリプションを選択します。
3. アクセス制御 **(IAM) >** 追加 **>** ロール割り当ての追加 をクリックします。
4. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*をクリックします。
5. *メンバー*タブで、次の手順を実行します。
   a. *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
   b. *メンバーを選択*をクリックします。



   c. アプリケーションの名前を検索します。

      次に例を示します。



a. アプリケーションを選択し、「選択」をクリックします。

b. *次へ*をクリックします。

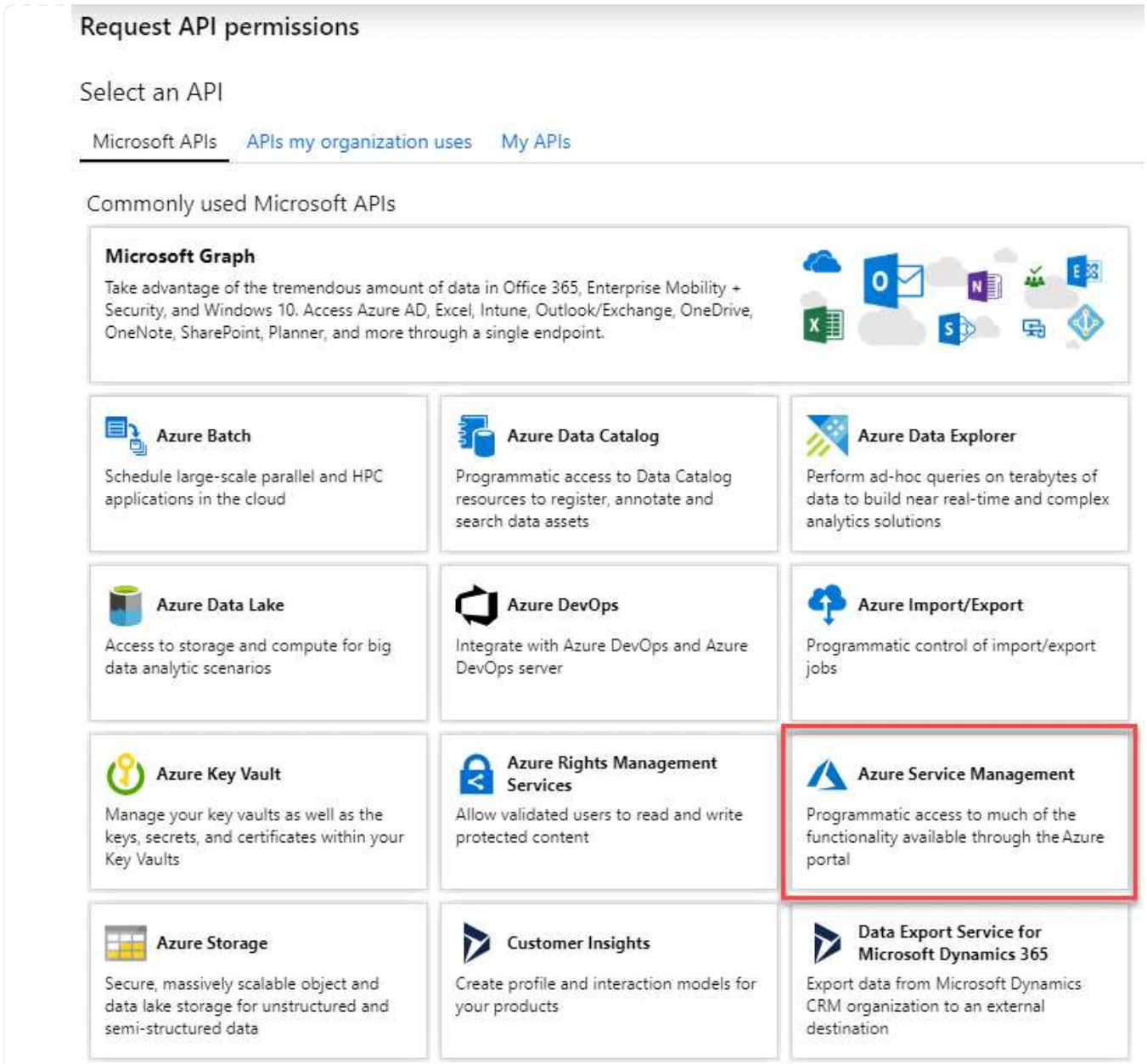
6. *レビュー+割り当て*をクリックします。

これで、サービス プリンシパルに、コンソール エージェントをデプロイするために必要な Azure アクセス許可が付与されました。

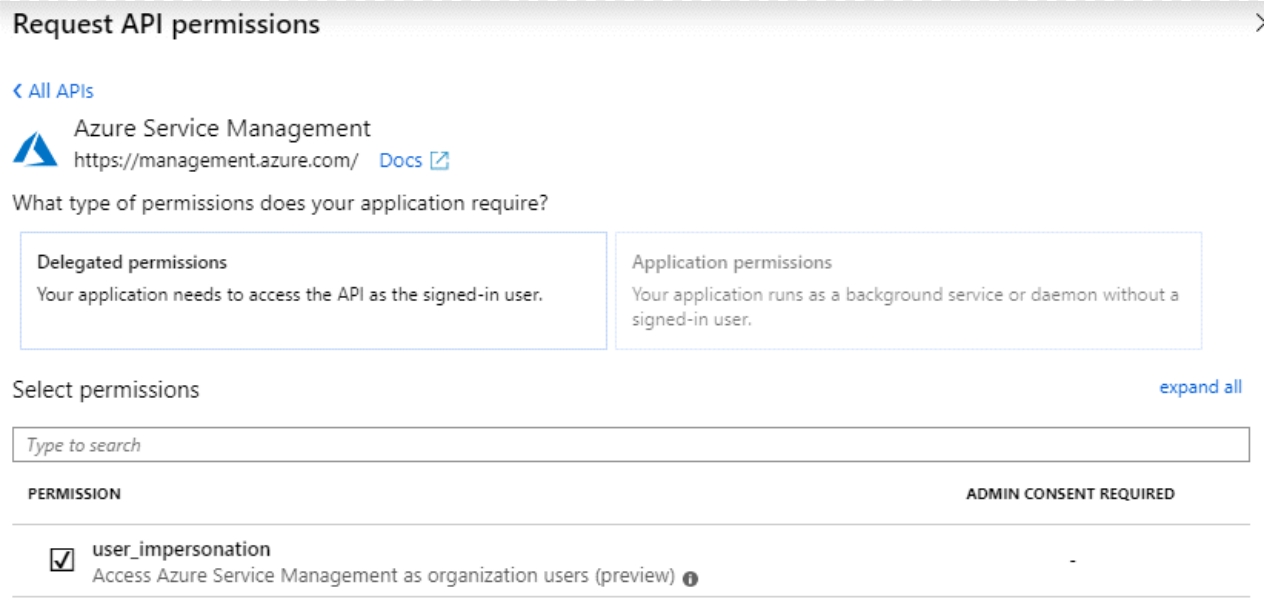
複数の Azure サブスクリプションのリソースを管理する場合は、サービス プリンシパルを各サブスクリプションにバインドする必要があります。たとえば、コンソールを使用すると、 Cloud Volumes ONTAPをデプロイするときに使用するサブスクリプションを選択できます。

Windows Azure サービス管理 **API** 権限を追加する

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. **API** 権限 **>** 権限の追加 を選択します。
3. **Microsoft API** の下で、**Azure Service Management** を選択します。

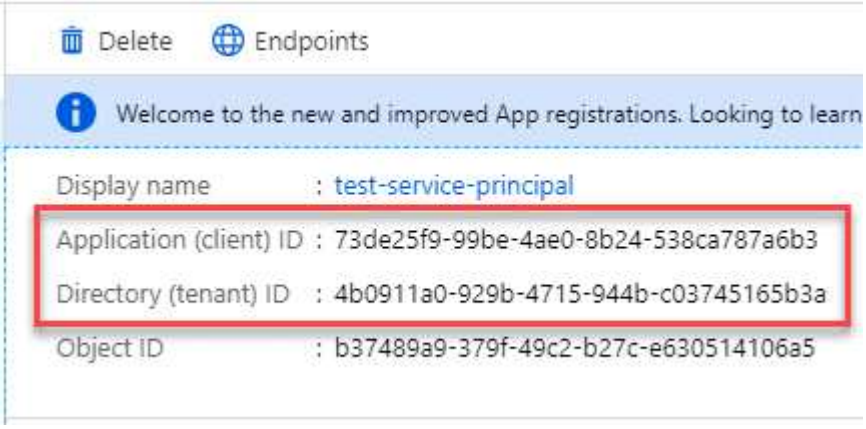


4. 組織ユーザーとして **Azure** サービス管理にアクセスする を選択し、権限の追加 を選択します。



アプリケーションのアプリケーション**ID**とディレクトリ**ID**を取得します

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. アプリケーション **(**クライアント**) ID** と ディレクトリ **(**テナント**) ID** をコピーします。



Azure アカウントをコンソールに追加するときは、アプリケーションのアプリケーション (クライアント) ID とディレクトリ (テナント) ID を指定する必要があります。コンソールは ID を使用してプログラムでサインインします。

クライアントシークレットを作成する

1. **Microsoft Entra ID** サービスを開きます。
2. *アプリ登録*を選択し、アプリケーションを選択します。
3. *証明書とシークレット > 新しいクライアント シークレット*を選択します。
4. シークレットの説明と期間を指定します。
5. *追加*を選択します。
6. クライアント シークレットの値をコピーします。

結果

これでサービス プリンシパルが設定され、アプリケーション (クライアント) ID、ディレクトリ (テナント) ID、およびクライアント シークレットの値がコピーされているはずです。コンソール エージェントを作成するときに、この情報をコンソールに入力する必要があります。

ステップ**4:** コンソールエージェントを作成する

NetApp Consoleから直接コンソール エージェントを作成します。

タスク概要

- コンソールからコンソール エージェントを作成すると、既定の構成を使用して Azure に仮想マシンがデプロイされます。コンソール エージェントを作成した後、CPU や RAM が少ない小さな VM インスタンスに切り替えないでください。"コンソールエージェントのデフォルト構成について学習します" 。
- コンソールはコンソール エージェントを展開するときに、カスタム ロールを作成し、それをコンソール エージェント VM に割り当てます。このロールには、コンソール エージェントが Azure リソースを管理できるようにする権限が含まれています。後続のリリースで新しい権限が追加されるので、ロールが最新の状態に保たれていることを確認する必要があります。"コンソールエージェントのカスタムロールの詳細" 。

開始する前に

次のものが必要です:

- Azure サブスクリプション。
- 選択した Azure リージョン内の VNet とサブネット。
- 組織ですべての送信インターネット トラフィックにプロキシが必要な場合のプロキシ サーバーの詳細:
  - IPアドレス
  - Credentials
  - HTTPS証明書
- コンソール エージェント仮想マシンにその認証方法を使用する場合は、SSH 公開キー。認証方法のもう 1 つのオプションは、パスワードを使用することです。

  "Azure の Linux VM への接続について学習します"

- コンソールエージェント用のAzureロールをコンソールが自動的に作成しないようにするには、独自のロールを作成する必要があります。"このページのポリシーを使用する" 。

  これらの権限は、コンソール エージェント自体に適用されます。これは、コンソール エージェント VM を展開するために以前に設定した権限セットとは異なります。

手順

1. *管理 > エージェント*を選択します。
2. *概要*ページで、*エージェントのデプロイ > Azure*を選択します。
3. レビュー ページで、エージェントを展開するための要件を確認します。これらの要件についてはこのページの上部にも詳しく記載されています。
4. *仮想マシン認証*ページで、Azure のアクセス許可の設定方法に一致する認証オプションを選択します。
   - ログイン を選択して、必要な権限を持つ Microsoft アカウントにログインします。

     このフォームは Microsoft によって所有およびホストされています。資格情報がNetAppに提供されていません。

     すでに Azure アカウントにログインしている場合は、コンソールは自動的にそのアカウントを使用します。複数のアカウントをお持ちの場合は、正しいアカウントを使用していることを確認するために、最初にログアウトする必要がある場合があります。

   - 必要な権限を付与する Microsoft Entra サービス プリンシパルに関する情報を入力するには、**Active Directory** サービス プリンシパル を選択します。
     - アプリケーション（クライアント）ID
     - ディレクトリ（テナント）ID
     - クライアントシークレット

   サービスプリンシパルのこれらの値を取得する方法を学びます 。
5. 仮想マシン認証 ページで、Azure サブスクリプション、場所、新しいリソース グループまたは既存のリソース グループを選択し、作成するコンソール エージェント仮想マシンの認証方法を選択します。

   仮想マシンの認証方法は、パスワードまたは SSH 公開キーです。

   "Azure の Linux VM への接続について学習します"
6. *詳細*ページで、エージェントの名前を入力し、タグを指定して、コンソールで必要な権限を持つ新しいロールを作成するか、または既存のロールを選択するかを選択します。"必要な権限" 。

   このロールに関連付けられた Azure サブスクリプションを選択できることに注意してください。選択した各サブスクリプションは、そのサブスクリプション内のリソースを管理するためのコンソール エージェント権限を付与します (たとえば、 Cloud Volumes ONTAP)。
7. ネットワーク ページで、VNet とサブネットを選択し、パブリック IP アドレスを有効にするかどうかを選択し、必要に応じてプロキシ構成を指定します。
   - セキュリティ グループ ページで、新しいセキュリティ グループを作成するか、必要な受信ルールと送信ルールを許可する既存のセキュリティ グループを選択するかを選択します。

     "Azure のセキュリティ グループ ルールを表示する" 。
8. 選択内容を確認して、セットアップが正しいことを確認します。
   a. エージェント構成の検証 チェック ボックスはデフォルトでオンになっており、展開時にコンソールによってネットワーク接続要件が検証されます。コンソールがエージェントの展開に失敗した場合、ト

ラブルシューティングに役立つレポートが提供されます。デプロイメントが成功した場合、レポートは提供されません。

まだ使用している場合は"以前のエンドポイント"エージェントのアップグレードに使用すると、検証が失敗し、エラーが発生します。これを回避するには、チェックボックスをオフにして検証チェックをスキップします。

9. *追加*を選択します。

コンソールは約 10 分でエージェントを準備します。プロセスが完了するまでこのページに留まります。

結果

プロセスが完了すると、コンソール エージェントはコンソールから使用できるようになります。

デプロイメントが失敗した場合は、コンソールからレポートとログをダウンロードして、問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

コンソール エージェントを作成したのと同じ Azure アカウントに Azure Blob ストレージがある場合は、システム ページに Azure Blob ストレージが自動的に表示されます。 "NetApp Consoleから Azure Blob ストレージを管理する方法を学びます"

Azure Marketplace からコンソール エージェントを作成する

Azure Marketplace から直接、Azure にコンソール エージェントを作成できます。Azure Marketplace からコンソール エージェントを作成するには、ネットワークを設定し、Azure のアクセス許可を準備し、インスタンスの要件を確認してから、コンソール エージェントを作成する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- レビュー"コンソールエージェントの制限"。

ステップ**1:** ネットワークを設定する

コンソール エージェントをインストールする予定のネットワークの場所が次の要件をサポートしていることを確認します。これらの要件により、コンソール エージェントはハイブリッド クラウド内のリソースを管理できるようになります。

Azureリージョン

Cloud Volumes ONTAPを使用する場合、コンソールエージェントは、管理するCloud Volumes ONTAPシステムと同じAzureリージョン、または "Azure リージョン ペア"Cloud Volumes ONTAPシステム用。この要件により、 Cloud Volumes ONTAPとそれに関連付けられたストレージ アカウント間で Azure Private Link 接続が使用されるようになります。

"Cloud Volumes ONTAP がAzure Private Link を使用する方法を学ぶ"

VNetとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VNet とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	Azure パブリック リージョン内のリソースを管理します。
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	Azure China リージョンのリソースを管理します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェア アップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。

エンドポイント	目的
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォール ルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インター

ネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、ネットワーク要件を実装します。

ステップ**2: VM**の要件を確認する

コンソール エージェントを作成するときは、次の要件を満たす仮想マシンの種類を選択します。

CPU

8コアまたは8vCPU

RAM

32 GB

Azure VM サイズ

CPU と RAM の要件を満たすインスタンス タイプ。 NetAppStandard_D8s_v3 を推奨します。

ステップ**3:** 権限を設定する

権限は次の方法で付与できます。

- オプション 1: システム割り当てマネージド ID を使用して、Azure VM にカスタム ロールを割り当てます。
- オプション 2: 必要な権限を持つ Azure サービス プリンシパルの資格情報をコンソールに提供します。

コンソールの権限を設定するには、次の手順に従います。

カスタムロール

Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

手順

1. 独自のホストにソフトウェアを手動でインストールする予定の場合は、カスタム ロールを通じて必要な Azure アクセス許可を提供できるように、VM でシステム割り当てマネージド ID を有効にします。

   "Microsoft Azure ドキュメント: Azure ポータルを使用して VM 上の Azure リソースのマネージド ID を構成する"

2. の内容をコピーします"コネクタのカスタムロール権限"JSON ファイルに保存します。
3. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

   NetApp Consoleで使用する各 Azure サブスクリプションの ID を追加する必要があります。

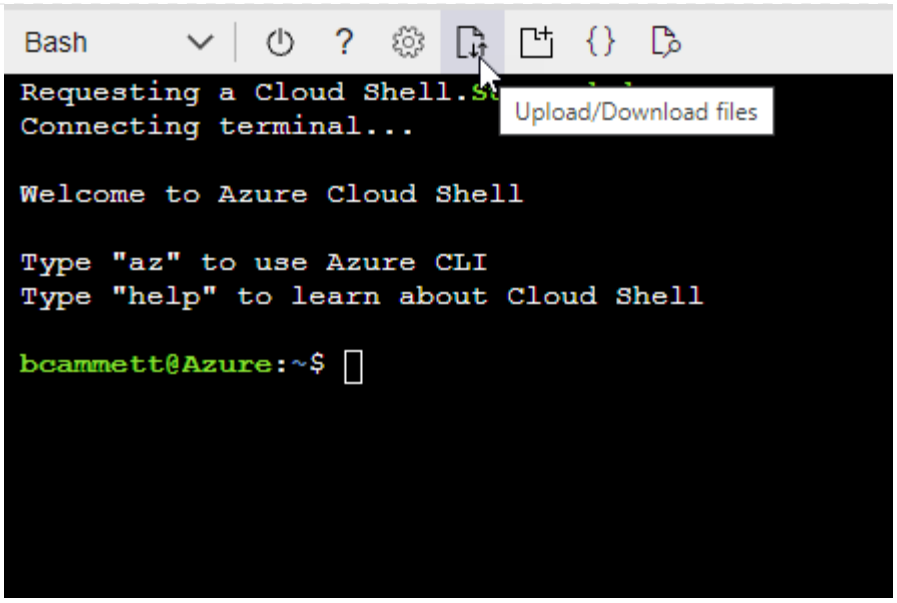
   例

   ```
   "AssignableScopes": [
   "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
   "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
   "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
   ]
   ```

4. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

   次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

   a. 始める "Azure クラウド シェル"Bash 環境を選択します。
   b. JSON ファイルをアップロードします。



c. Azure CLI を使用してカスタム ロールを作成します。

```
az role definition create --role-definition agent_Policy.json
```

サービスプリンシパル

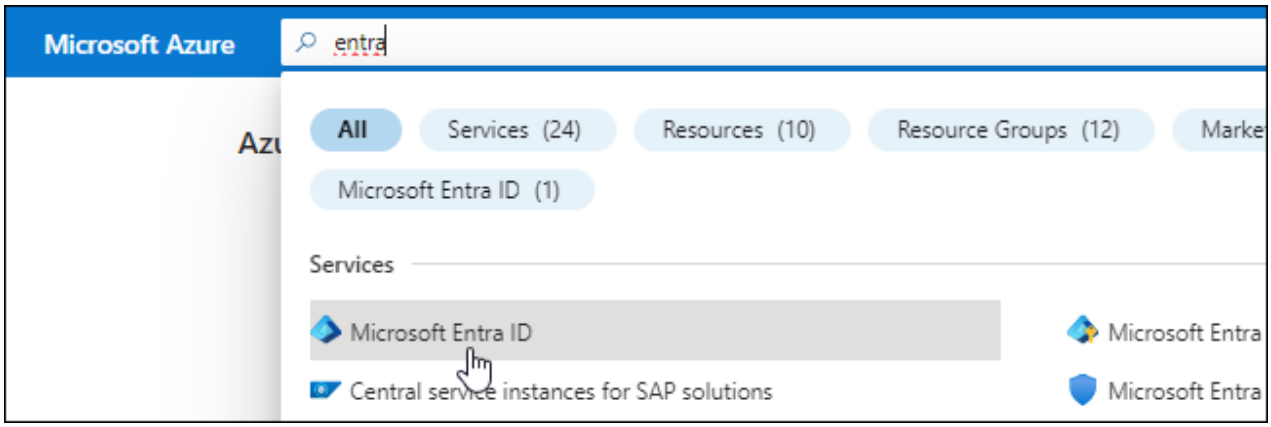
Microsoft Entra ID でサービス プリンシパルを作成して設定し、コンソールに必要な Azure 資格情報を取得します。

ロールベースのアクセス制御用の **Microsoft Entra** アプリケーションを作成する

1. Azure で Active Directory アプリケーションを作成し、そのアプリケーションをロールに割り当てるためのアクセス許可があることを確認します。

   詳細については、 "Microsoft Azure ドキュメント: 必要な権限"

2. Azure ポータルから、**Microsoft Entra ID** サービスを開きます。



3. メニューで*アプリ登録*を選択します。
4. *新規登録*を選択します。

5. アプリケーションの詳細を指定します。
   - 名前: アプリケーションの名前を入力します。
   - アカウント タイプ: アカウント タイプを選択します (いずれのタイプもNetApp Consoleで使用できます)。
   - リダイレクト **URI**: このフィールドは空白のままにすることができます。
6. *登録*を選択します。

   AD アプリケーションとサービス プリンシパルを作成しました。

アプリケーションをロールに割り当てる

1. カスタム ロールを作成します。

   Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

   a. の内容をコピーします"コンソールエージェントのカスタムロール権限"JSON ファイルに保存します。
   b. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

      ユーザーがCloud Volumes ONTAPシステムを作成する各 Azure サブスクリプションの ID を追加する必要があります。

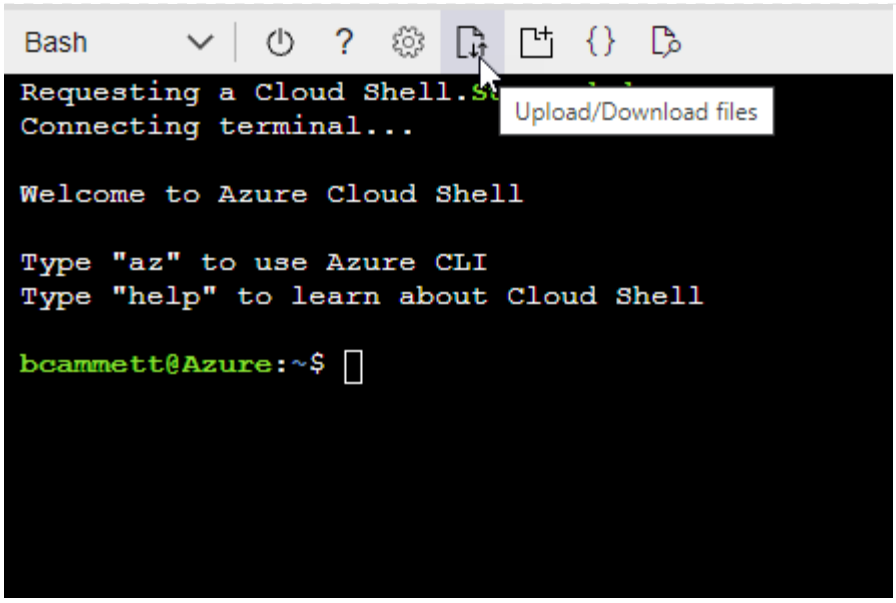
      例

      ```
      "AssignableScopes": [
      "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
      "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
      "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
      ]
      ```

   c. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

      次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

      - 始める "Azure クラウド シェル"Bash 環境を選択します。
      - JSON ファイルをアップロードします。



- Azure CLI を使用してカスタム ロールを作成します。

```
az role definition create --role-definition agent_Policy.json
```

これで、コンソール エージェント仮想マシンに割り当てることができる、コンソール オペレーターと呼ばれるカスタム ロールが作成されます。

2. アプリケーションをロールに割り当てます。

a. Azure ポータルから、サブスクリプション サービスを開きます。

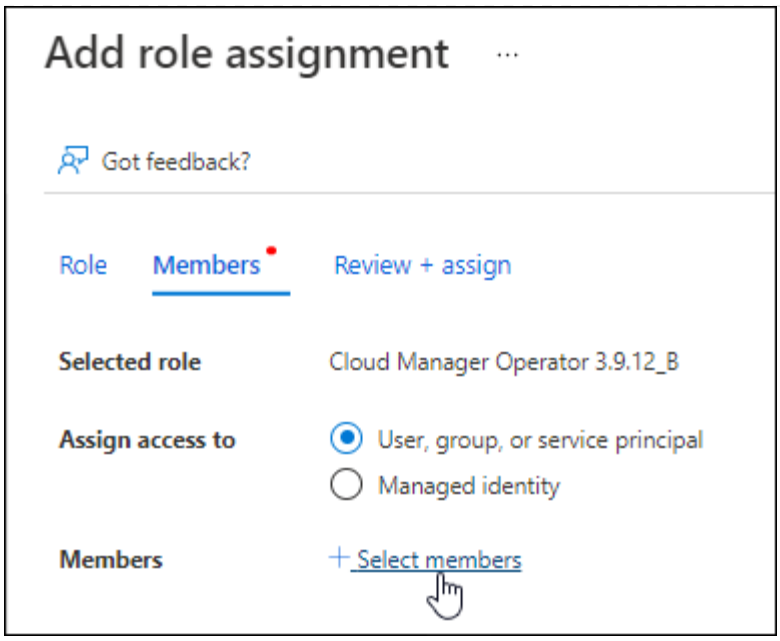
b. サブスクリプションを選択します。

c. アクセス制御 **(IAM) >** 追加 **>** ロール割り当ての追加 を選択します。

d. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。

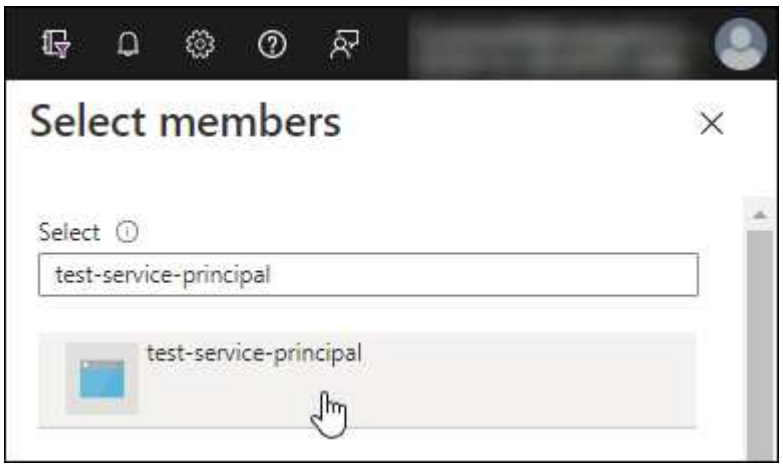
e. *メンバー*タブで、次の手順を実行します。

- *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
- *メンバーを選択*を選択します。



- アプリケーションの名前を検索します。

  次に例を示します。



- アプリケーションを選択し、[選択] を選択します。
- *次へ*を選択します。

f. *レビュー + 割り当て*を選択します。

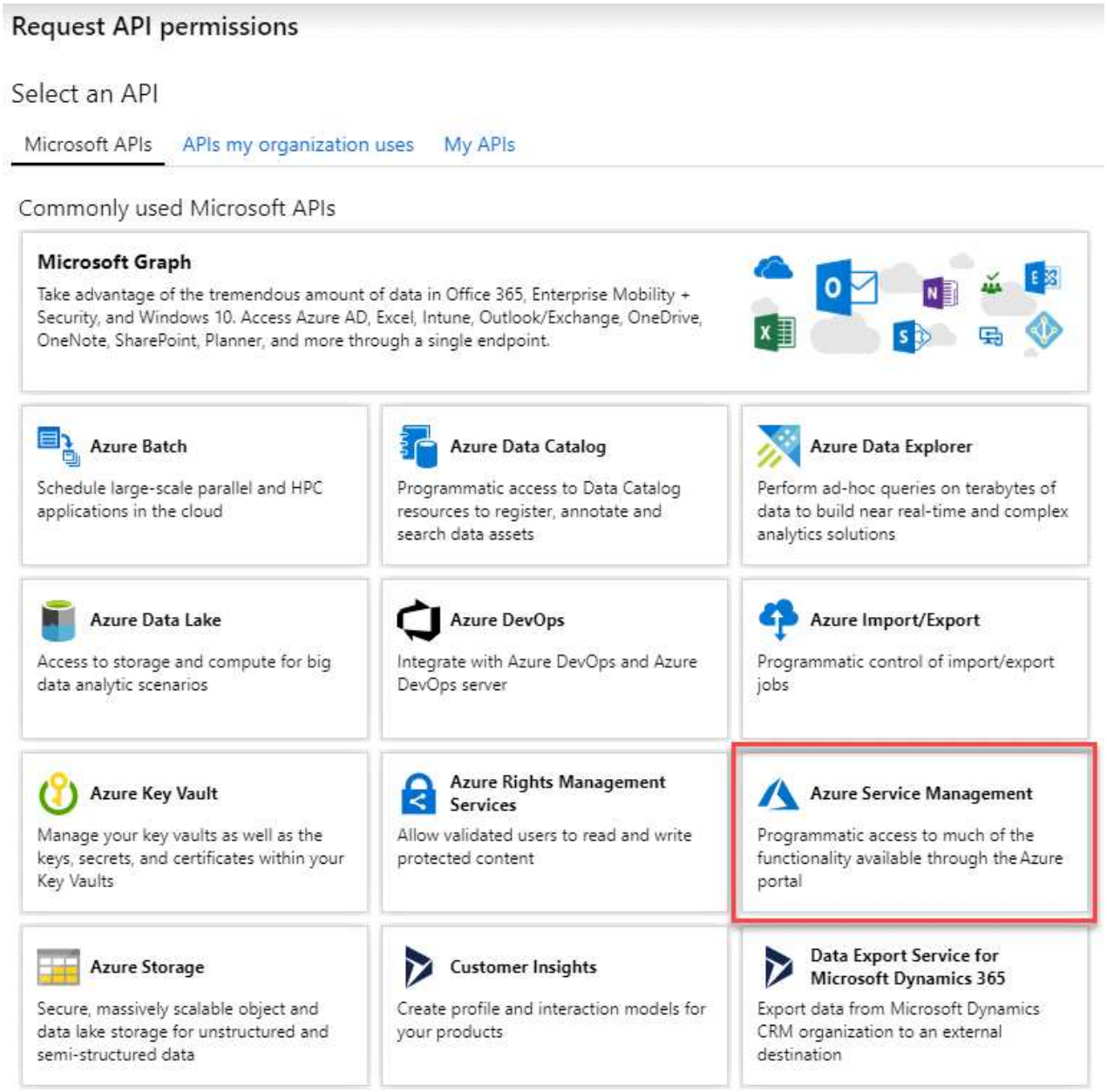
これで、サービス プリンシパルに、コンソール エージェントをデプロイするために必要な Azure アクセス許可が付与されました。

複数の Azure サブスクリプションからCloud Volumes ONTAPをデプロイする場合は、サービス プリンシパルを各サブスクリプションにバインドする必要があります。 NetApp Consoleでは、 Cloud Volumes ONTAP をデプロイするときに使用するサブスクリプションを選択できます。

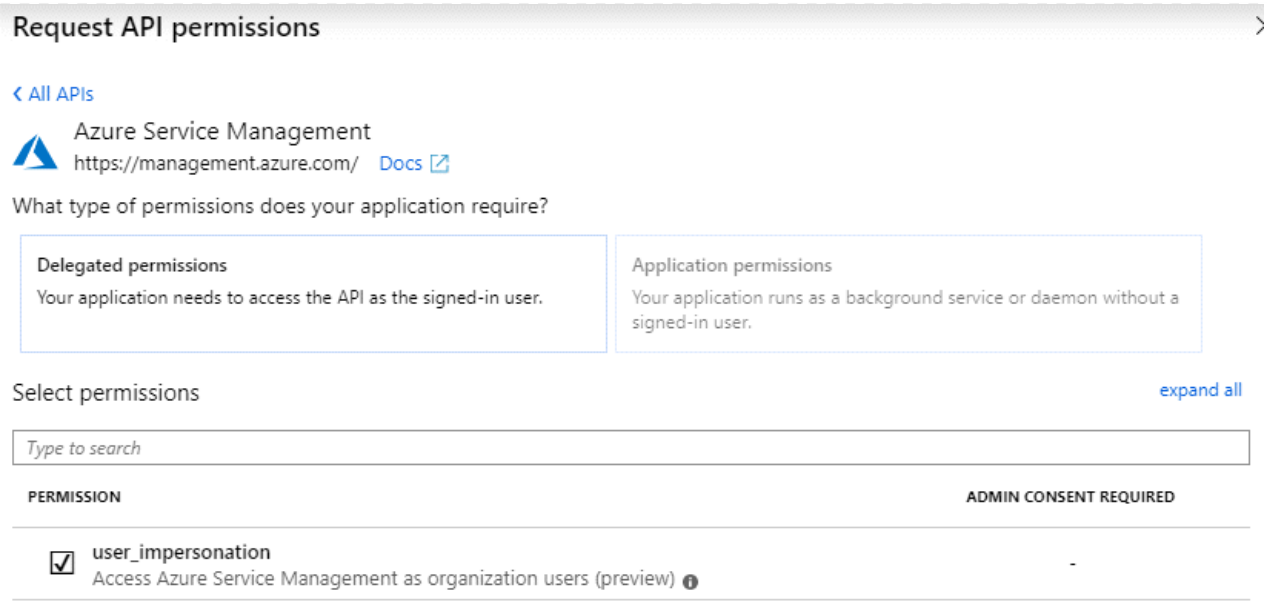
Windows Azure サービス管理 **API** 権限を追加する

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. **API** 権限 **>** 権限の追加 を選択します。

3. **Microsoft API** の下で、**Azure Service Management** を選択します。

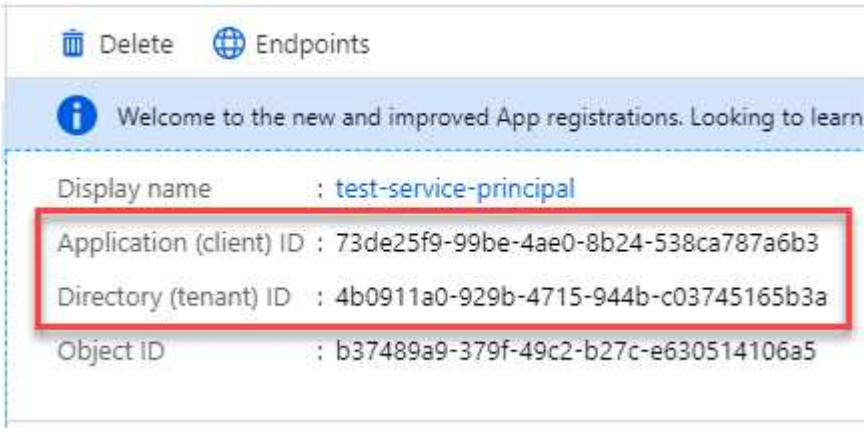


4. 組織ユーザーとして **Azure** サービス管理にアクセスする を選択し、権限の追加 を選択します。



アプリケーションのアプリケーション**ID**とディレクトリ**ID**を取得します

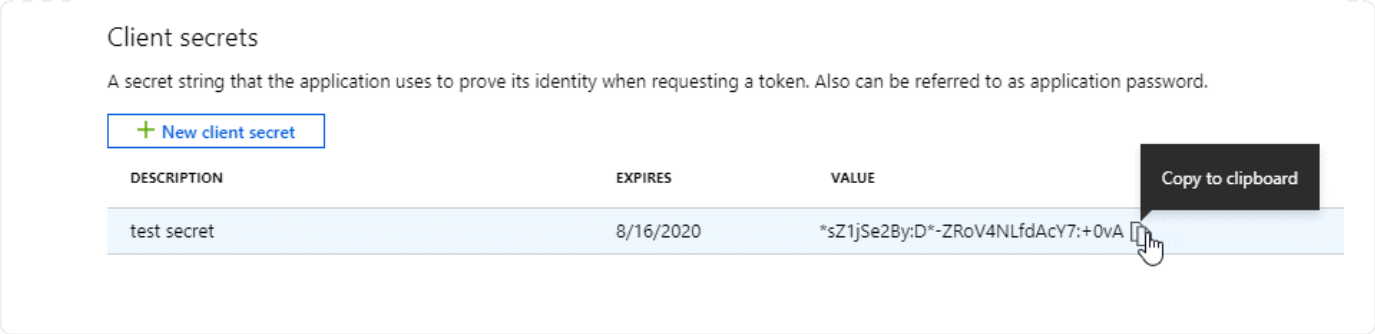
1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. アプリケーション **(**クライアント**) ID** と ディレクトリ **(**テナント**) ID** をコピーします。



Azure アカウントをコンソールに追加するときは、アプリケーションのアプリケーション (クライアント) ID とディレクトリ (テナント) ID を指定する必要があります。コンソールは ID を使用してプログラムでサインインします。

クライアントシークレットを作成する

1. **Microsoft Entra ID** サービスを開きます。
2. *アプリ登録*を選択し、アプリケーションを選択します。
3. *証明書とシークレット > 新しいクライアント シークレット*を選択します。
4. シークレットの説明と期間を指定します。
5. *追加*を選択します。
6. クライアント シークレットの値をコピーします。



ステップ**4:** コンソールエージェントを作成する

Azure Marketplace からコンソール エージェントを直接起動します。

タスク概要

Azure Marketplace からコンソール エージェントを作成すると、既定の構成で仮想マシンが設定されます。"コンソールエージェントのデフォルト構成について学習します" 。

開始する前に

次のものが必要です:

- Azure サブスクリプション。
- 選択した Azure リージョン内の VNet とサブネット。
- 組織ですべての送信インターネット トラフィックにプロキシが必要な場合のプロキシ サーバーの詳細:
  - IPアドレス
  - Credentials
  - HTTPS証明書
- コンソール エージェント仮想マシンにその認証方法を使用する場合は、SSH 公開キー。認証方法のもう 1 つのオプションは、パスワードを使用することです。

  "Azure の Linux VM への接続について学習します"

- コンソールエージェント用のAzureロールをコンソールが自動的に作成しないようにするには、独自のロールを作成する必要があります。"このページのポリシーを使用する" 。

  これらの権限は、コンソール エージェント インスタンス自体に適用されます。これは、コンソール エージェント VM を展開するために以前に設定した権限セットとは異なります。

手順

1. Azure Marketplace のNetApp Consoleエージェント VM ページに移動します。

   "商用リージョン向けの Azure Marketplace ページ"

2. *今すぐ入手*を選択し、*続行*を選択します。
3. Azure ポータルから [作成] を選択し、手順に従って仮想マシンを構成します。

   VM を構成する際には、次の点に注意してください。

- **VM** サイズ: CPU と RAM の要件を満たす VM サイズを選択します。 Standard_D8s_v3 をお勧めします。
- ディスク: コンソール エージェントは、HDD ディスクまたは SSD ディスクのいずれでも最適に動作します。
- ネットワーク セキュリティ グループ: コンソール エージェントには、SSH、HTTP、および HTTPS を使用した受信接続が必要です。

  "Azure のセキュリティ グループ ルールを表示する" 。

- ID*: 管理 の下で、システム割り当てマネージド **ID** を有効にする を選択します。

  この設定は重要です。マネージド ID を使用すると、コンソール エージェント仮想マシンは資格情報を提供せずに Microsoft Entra ID に対して自身を識別できるためです。 "Azure リソースのマネージド ID の詳細" 。

4. 確認 **+** 作成 ページで選択内容を確認し、作成 を選択してデプロイを開始します。

   Azure は指定された設定で仮想マシンをデプロイします。約 10 分以内に仮想マシンとコンソール エージェント ソフトウェアが実行されるはずです。

   インストールが失敗した場合は、トラブルシューティングに役立つログとレポートを表示できます。"インストールの問題をトラブルシューティングする方法を学びます。"

5. コンソール エージェント仮想マシンに接続しているホストから Web ブラウザを開き、次の URL を入力します。

   https://ipaddress

6. ログイン後、コンソール エージェントを設定します。
   a. コンソール エージェントに関連付けるコンソール組織を指定します。
   b. システムの名前を入力します。
   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      コンソールを標準モードで使用するには、制限モードを無効にしておきます。安全な環境があり、このアカウントをコンソールのバックエンド サービスから切断する場合にのみ、制限モードを有効にする必要があります。もしそうなら、"制限モードでコンソールを開始するには、以下の手順に従ってください。" 。

   d. *始めましょう*を選択します。

結果

これで、コンソール エージェントがインストールされ、コンソール組織で設定されました。

コンソール エージェントを作成したのと同じ Azure サブスクリプションに Azure Blob ストレージがある場合は、システム ページに Azure Blob ストレージ システムが自動的に表示されます。 "コンソールから Azure Blob ストレージを管理する方法を学びます"

ステップ**5:** コンソールエージェントに権限を付与する

コンソール エージェントを作成したので、以前に設定した権限をエージェントに付与する必要があります。権限を付与すると、コンソール エージェントは Azure 内のデータとストレージ インフラストラクチャを管理できるようになります。

カスタムロール

Azure ポータルに移動し、1 つ以上のサブスクリプションのコンソール エージェント仮想マシンに Azure カスタム ロールを割り当てます。

手順

1. Azure ポータルから サブスクリプション サービスを開き、サブスクリプションを選択します。

   サブスクリプション サービスからロールを割り当てることが重要です。これは、サブスクリプション レベルでのロール割り当ての範囲を指定するためです。 *scope* は、アクセスが適用されるリソースのセットを定義します。別のレベル (たとえば、仮想マシン レベル) でスコープを指定すると、NetApp Console内からアクションを完了する機能に影響します。

   "Microsoft Azure ドキュメント: Azure RBAC のスコープを理解する"

2. アクセス制御 **(IAM)** > 追加 > ロール割り当ての追加 を選択します。
3. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。

   コンソール オペレーターは、ポリシーで提供されるデフォルト名です。ロールに別の名前を選択した場合は、代わりにその名前を選択します。

4. *メンバー*タブで、次の手順を実行します。
   a. マネージド **ID** へのアクセスを割り当てます。
   b. *メンバーの選択*を選択し、コンソール エージェント仮想マシンが作成されたサブスクリプションを選択し、*マネージド ID*の下で*仮想マシン*を選択して、コンソール エージェント仮想マシンを選択します。
   c. *選択*を選択します。
   d. *次へ*を選択します。
   e. *レビュー + 割り当て*を選択します。
   f. 追加の Azure サブスクリプションのリソースを管理する場合は、そのサブスクリプションに切り替えて、これらの手順を繰り返します。

次の手順

に行く "NetApp Console"コンソール エージェントの使用を開始します。

サービスプリンシパル

手順

1. *管理 > 資格情報*を選択します。
2. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: **Microsoft Azure >** エージェント を選択します。
   b. 資格情報の定義: 必要な権限を付与する Microsoft Entra サービス プリンシパルに関する情報を入力します。
      - アプリケーション（クライアント）ID
      - ディレクトリ（テナント）ID

- クライアントシークレット

c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。

d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

結果

これで、コンソールに、ユーザーに代わって Azure でアクションを実行するために必要なアクセス許可が付与されました。

Azure にコンソール エージェントを手動でインストールする

独自の Linux ホストにコンソール エージェントを手動でインストールするには、ホストの要件を確認し、ネットワークを設定し、Azure のアクセス許可を準備し、コンソール エージェントをインストールして、準備したアクセス許可を付与する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ホストの要件を確認する

コンソール エージェント ソフトウェアは、特定のオペレーティング システム要件、RAM 要件、ポート要件などを満たすホスト上で実行する必要があります。

コンソール エージェントは、19000 ～ 19200 の UID と GID の範囲を予約します。この範囲は固定されており、変更することはできません。ホスト上のサードパーティ ソフトウェアがこの範囲内の UID または GID を使用している場合、エージェントのインストールは失敗します。NetApp、競合を回避するためにサードパーティ ソフトウェアがインストールされていないホストの使用を推奨しています。

専用ホスト

コンソール エージェントには専用のホストが必要です。次のサイズ要件を満たすアーキテクチャであれば、どれでもサポートされます。

- CPU: 8コアまたは8vCPU
- メモリ: 32 GB
- ディスク容量: ホストには 165 GB が推奨され、パーティション要件は次のとおりです。
  - `/opt`: 120 GiBの空き容量が必要です

    エージェントは `/opt`インストールするには `/opt/application/netapp`ディレクトリとその内容。

  - `/var`: 40 GiBの空き容量が必要です

    コンソールエージェントにはこのスペースが必要です `/var`Podman または Docker は、このディレクトリ内にコンテナを作成するように設計されているためです。具体的には、`/var/lib/containers/storage`ディレクトリと `/var/lib/docker`Docker用。このスペースでは外部マウ

ントまたはシンボリックリンクは機能しません。

Azure VM サイズ

CPU と RAM の要件を満たすインスタンス タイプ。 NetAppStandard_D8s_v3 を推奨します。

ハイパーバイザー

サポートされているオペレーティング システムを実行することが認定されているベア メタルまたはホスト型ハイパーバイザーが必要です。

オペレーティングシステムとコンテナの要件

コンソールを標準モードまたは制限モードで使用する場合、コンソール エージェントは次のオペレーティング システムでサポートされます。エージェントをインストールする前に、コンテナ オーケストレーション ツールが必要です。

オペレーティング システム	サポートされる**OS** バージョン	サポートされているエージェントのバージョン	必要なコンテナツール	**SELinux**
Red Hat Enterprise Linux		9.6 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	4.0.0 以降、コンソールが標準モードまたは制限モード	Podman バージョン 5.4.0 と podman-compose 1.5.0。 Podman の構成要件を表示する 。

オペレーティングシステム	サポートされるOSバージョン	サポートされているエージェントのバージョン	必要なコンテナツール	SELinux
強制モードまたは許可モードでサポートされます		9.1～9.4 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.9.4 と podman-compose 1.5.0。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます		8.6～8.10 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.6.1 または 4.9.4 と podman-compose 1.0.6。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます	Ubuntu		24.04 LTS	3.9.45 以降、NetApp Consoleが標準モードまたは制限モード
Docker エンジン 23.06 から 28.0.0。	サポート対象外		22.04 LTS	3.9.50以降

ステップ**2: Podman**または**Docker Engine**をインストールする

オペレーティング システムに応じて、エージェントをインストールする前に Podman または Docker Engine のいずれかが必要になります。

- Red Hat Enterprise Linux 8 および 9 には Podman が必要です。

  サポートされているPodmanのバージョンを表示する 。

- Ubuntu には Docker Engine が必要です。

  サポートされている Docker エンジンのバージョンを表示する 。

例 **2.** 手順

ポッドマン

Podman をインストールして設定するには、次の手順に従います。

- podman.socket サービスを有効にして起動します
- Python3をインストールする
- podman-compose パッケージ バージョン 1.0.6 をインストールします。
- podman-composeをPATH環境変数に追加する
- Red Hat Enterprise Linux を使用している場合は、Podman バージョンが CNI ではなく Netavark Aardvark DNS を使用していることを確認してください。

DNS ポートの競合を避けるために、エージェントをインストールした後、aardvark-dns ポート (デフォルト: 53) を調整します。指示に従ってポートを構成します。

手順

1. ホストに podman-docker パッケージがインストールされている場合は削除します。

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Podman をインストールします。

   Podman は、公式の Red Hat Enterprise Linux リポジトリから入手できます。

   a. Red Hat Enterprise Linux 9.6 の場合:

   ```
   sudo dnf install podman-5:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   b. Red Hat Enterprise Linux 9.1 から 9.4 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   c. Red Hat Enterprise Linux 8 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

<version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

3. podman.socket サービスを有効にして起動します。

```
sudo systemctl enable --now podman.socket
```

4. python3 をインストールします。

```
sudo dnf install python3
```

5. システムにまだインストールされていない場合は、EPEL リポジトリ パッケージをインストールします。

   podman-compose は、Extra Packages for Enterprise Linux (EPEL) リポジトリから入手できるため、この手順は必須です。

6. Red Hat Enterprise 9 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

   +

   a. podman-compose パッケージ 1.5.0 をインストールします。

```
sudo dnf install podman-compose-1.5.0
```

7. Red Hat Enterprise Linux 8 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

   b. podman-compose パッケージ 1.0.6 をインストールします。

```
sudo dnf install podman-compose-1.0.6
```

使用して `dnf install`コマンドは、PATH 環境変数に podman-compose を追加するための要件を満たしています。インストールコマンドは、すでにインストールされているpodman-composeを/usr/binに追加します。 `secure_path`ホスト上のオプション。

c. Red Hat Enterprise Linux 8 を使用している場合は、Podman バージョンが CNI ではなく Aardvark DNS を備えた NetAvark を使用していることを確認します。

i. 次のコマンドを実行して、networkBackend が CNI に設定されているかどうかを確認します。

```
podman info | grep networkBackend
```

ii. networkBackendが `CNI`、それを変更する必要があります `netavark`。

iii. インストール `netavark`そして `aardvark-dns`次のコマンドを使用します。

```
dnf install aardvark-dns netavark
```

iv. 開く `/etc/containers/containers.conf`ファイルを編集し、network_backend オプションを変更して、「cni」の代わりに「netavark」を使用します。

もし `/etc/containers/containers.conf`存在しない場合は、設定を変更してください `/usr/share/containers/containers.conf`。

v. podman を再起動します。

```
systemctl restart podman
```

vi. 次のコマンドを使用して、networkBackend が「netavark」に変更されていることを確認します。

```
podman info | grep networkBackend
```

Dockerエンジン

Docker のドキュメントに従って Docker Engine をインストールします。

手順

1. "Dockerからのインストール手順を見る"

   サポートされている Docker エンジン バージョンをインストールするには、手順に従ってください。最新バージョンはコンソールでサポートされていないため、インストールしないでください。

2. Docker が有効になっていて実行されていることを確認します。

```
sudo systemctl enable docker && sudo systemctl start docker
```

ステップ**3:** ネットワークを設定する

コンソール エージェントをインストールする予定のネットワークの場所が次の要件をサポートしていることを確認します。これらの要件を満たすことで、コンソール エージェントはハイブリッド クラウド環境内のリソースとプロセスを管理できるようになります。

Azureリージョン

Cloud Volumes ONTAPを使用する場合、コンソールエージェントは、管理するCloud Volumes ONTAPシステムと同じAzureリージョン、または "Azure リージョン ペア"Cloud Volumes ONTAPシステム用。この要件により、 Cloud Volumes ONTAPとそれに関連付けられたストレージ アカウント間で Azure Private Link 接続が使用されるようになります。

"Cloud Volumes ONTAP がAzure Private Link を使用する方法を学ぶ"

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

Webベースの**NetApp Console**を使用する際にコンピュータから接続されるエンドポイント

Web ブラウザからコンソールにアクセスするコンピュータは、複数のエンドポイントに接続できる必要があります。コンソール エージェントを設定し、コンソールを日常的に使用するには、コンソールを使用する必要があります。

"NetAppコンソールのネットワークを準備する" 。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	Azure パブリック リージョン内のリソースを管理します。
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	Azure China リージョンのリソースを管理します。

エンドポイント	目的
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

ステップ**4:** コンソールエージェントの展開権限を設定する

次のいずれかのオプションを使用して、コンソール エージェントに Azure 権限を付与する必要があります。

- オプション 1: システム割り当てマネージド ID を使用して、Azure VM にカスタム ロールを割り当てます。
- オプション 2: 必要なアクセス許可を持つ Azure サービス プリンシパルの資格情報をコンソール エージェントに提供します。

手順に従って、コンソール エージェントの権限を準備します。

コンソールエージェントの展開用のカスタムロールを作成する

Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

手順

1. 独自のホストにソフトウェアを手動でインストールする予定の場合は、カスタム ロールを通じて必要な Azure アクセス許可を提供できるように、VM でシステム割り当てマネージド ID を有効にします。

   "Microsoft Azure ドキュメント: Azure ポータルを使用して VM 上の Azure リソースのマネージド ID を構成する"

2. の内容をコピーします"コネクタのカスタムロール権限"JSON ファイルに保存します。
3. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

   NetApp Consoleで使用する各 Azure サブスクリプションの ID を追加する必要があります。

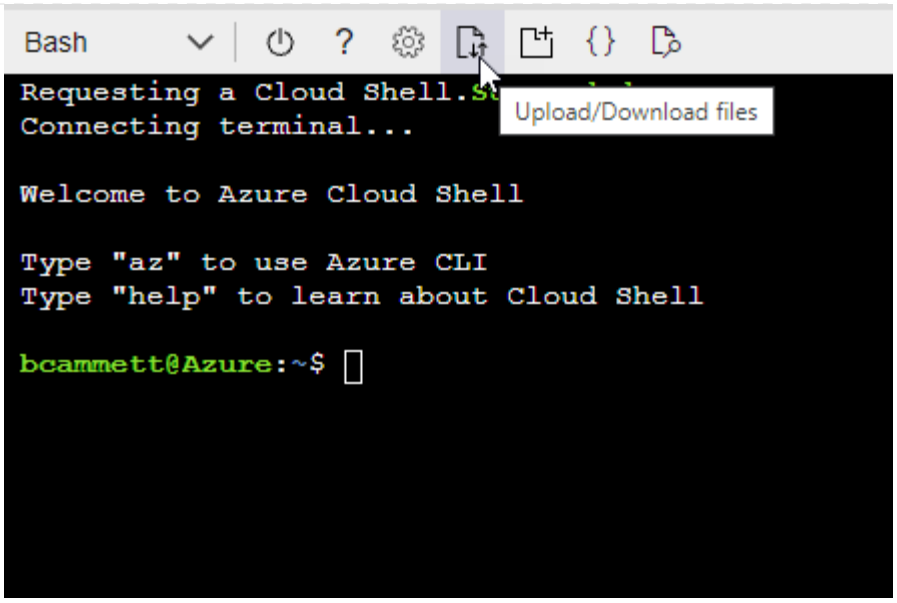
   例

   ```
   "AssignableScopes": [
   "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
   "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
   "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
   ]
   ```

4. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

   次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

   a. 始める "Azure クラウド シェル"Bash 環境を選択します。
   b. JSON ファイルをアップロードします。



c. Azure CLI を使用してカスタム ロールを作成します。

```
az role definition create --role-definition agent_Policy.json
```

サービスプリンシパル

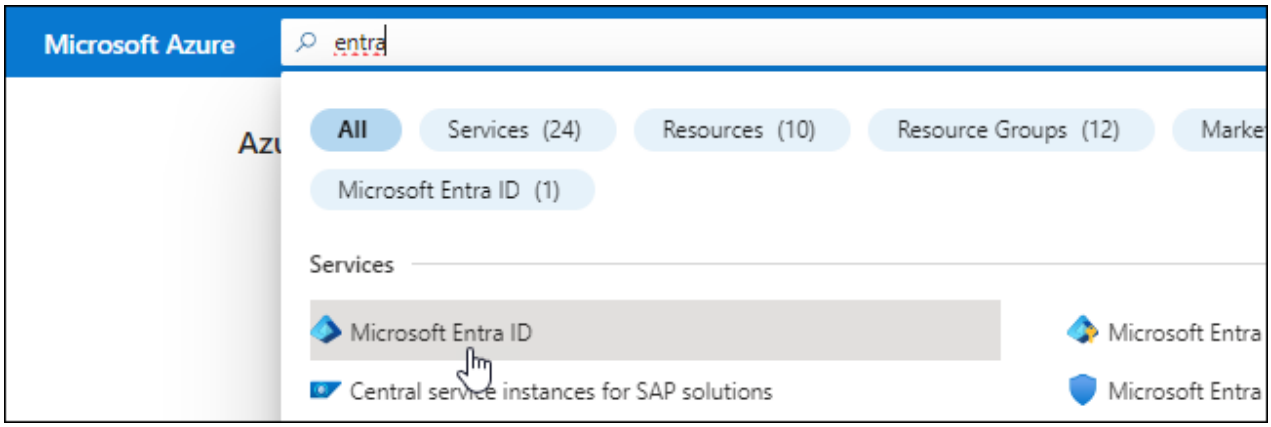
Microsoft Entra ID でサービス プリンシパルを作成して設定し、コンソール エージェントに必要な Azure 資格情報を取得します。

ロールベースのアクセス制御用の **Microsoft Entra** アプリケーションを作成する

1. Azure で Active Directory アプリケーションを作成し、そのアプリケーションをロールに割り当てるためのアクセス許可があることを確認します。

   詳細については、 "Microsoft Azure ドキュメント: 必要な権限"

2. Azure ポータルから、**Microsoft Entra ID** サービスを開きます。



3. メニューで*アプリ登録*を選択します。
4. *新規登録*を選択します。

5. アプリケーションの詳細を指定します。
   - 名前: アプリケーションの名前を入力します。
   - アカウント タイプ: アカウント タイプを選択します (いずれのタイプもNetApp Consoleで使用できます)。
   - リダイレクト **URI**: このフィールドは空白のままにすることができます。
6. *登録*を選択します。

   AD アプリケーションとサービス プリンシパルを作成しました。

アプリケーションをロールに割り当てる

1. カスタム ロールを作成します。

   Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

   a. の内容をコピーします"コンソールエージェントのカスタムロール権限"JSON ファイルに保存します。
   b. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

      ユーザーがCloud Volumes ONTAPシステムを作成する各 Azure サブスクリプションの ID を追加する必要があります。

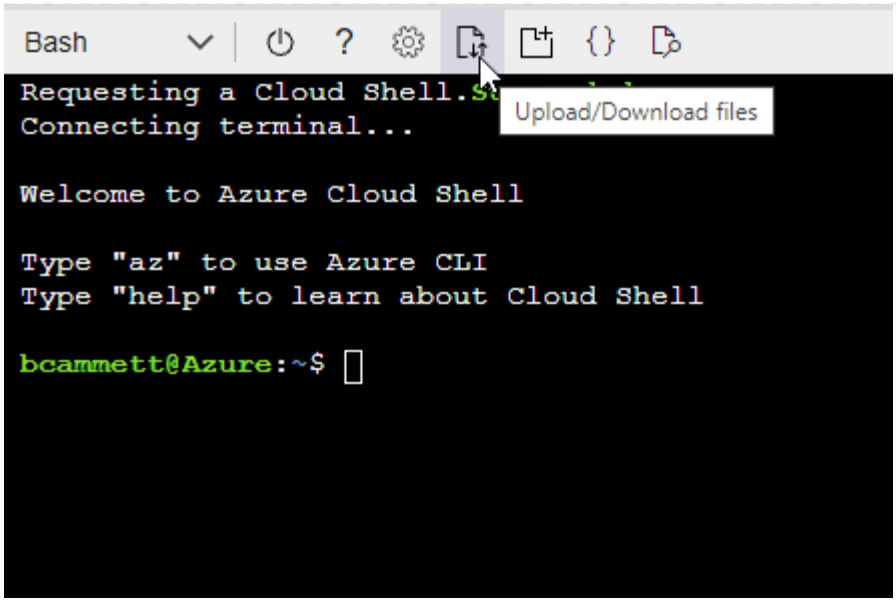
      例

      ```
      "AssignableScopes": [
      "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
      "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
      "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
      ]
      ```

   c. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

      次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

      - 始める "Azure クラウド シェル"Bash 環境を選択します。
      - JSON ファイルをアップロードします。



- Azure CLI を使用してカスタム ロールを作成します。

```
az role definition create --role-definition agent_Policy.json
```

これで、コンソール エージェント仮想マシンに割り当てることができる、コンソール オペレーターと呼ばれるカスタム ロールが作成されます。

2. アプリケーションをロールに割り当てます。

a. Azure ポータルから、サブスクリプション サービスを開きます。

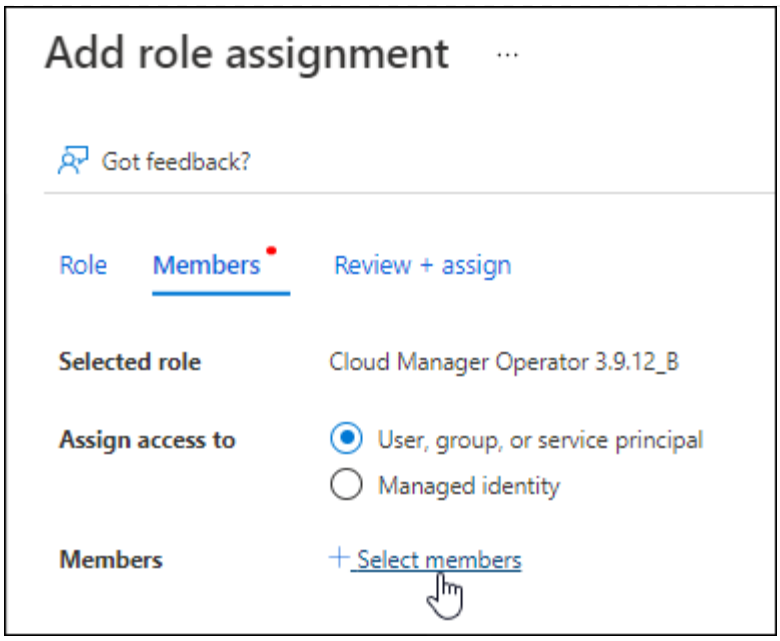
b. サブスクリプションを選択します。

c. アクセス制御 **(IAM) >** 追加 **>** ロール割り当ての追加 を選択します。

d. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。

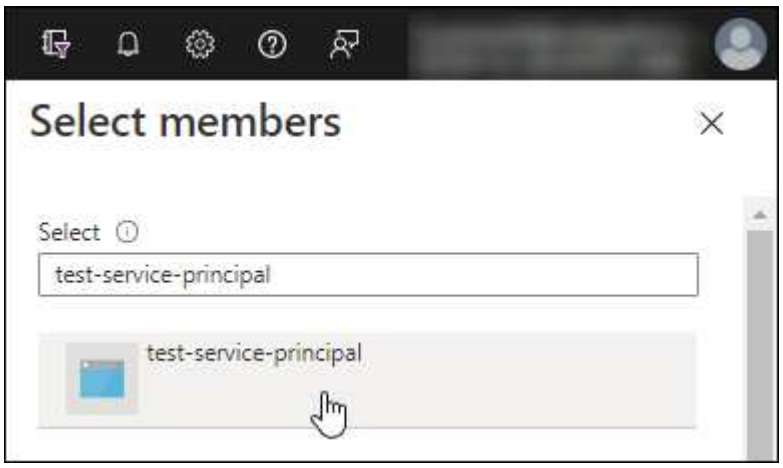
e. *メンバー*タブで、次の手順を実行します。

- *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
- *メンバーを選択*を選択します。



- アプリケーションの名前を検索します。

  次に例を示します。



- アプリケーションを選択し、[選択] を選択します。
- *次へ*を選択します。

f. *レビュー + 割り当て*を選択します。

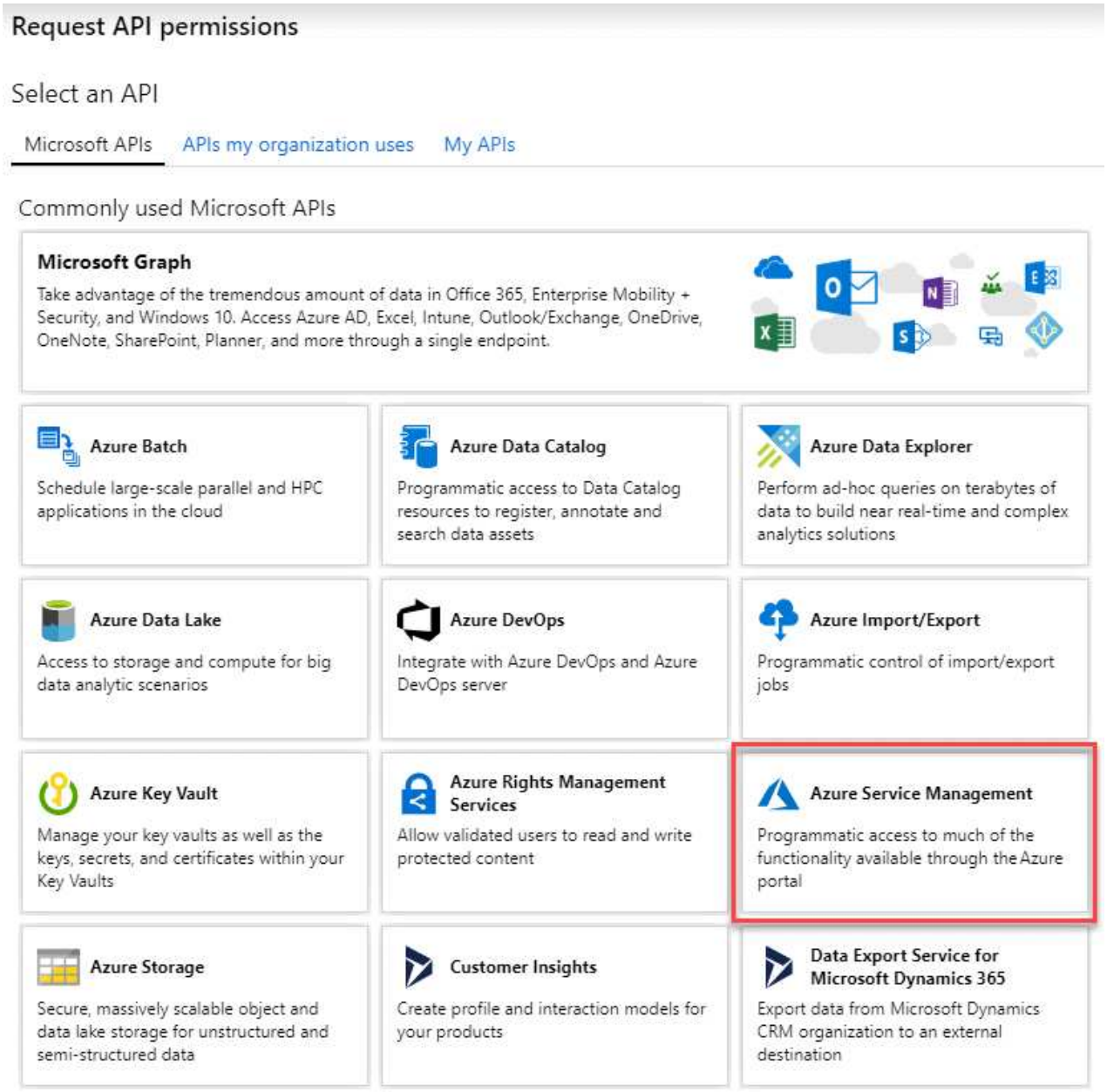
これで、サービス プリンシパルに、コンソール エージェントをデプロイするために必要な Azure アクセス許可が付与されました。

複数の Azure サブスクリプションからCloud Volumes ONTAPをデプロイする場合は、サービス プリンシパルを各サブスクリプションにバインドする必要があります。 NetApp Consoleでは、 Cloud Volumes ONTAP をデプロイするときに使用するサブスクリプションを選択できます。

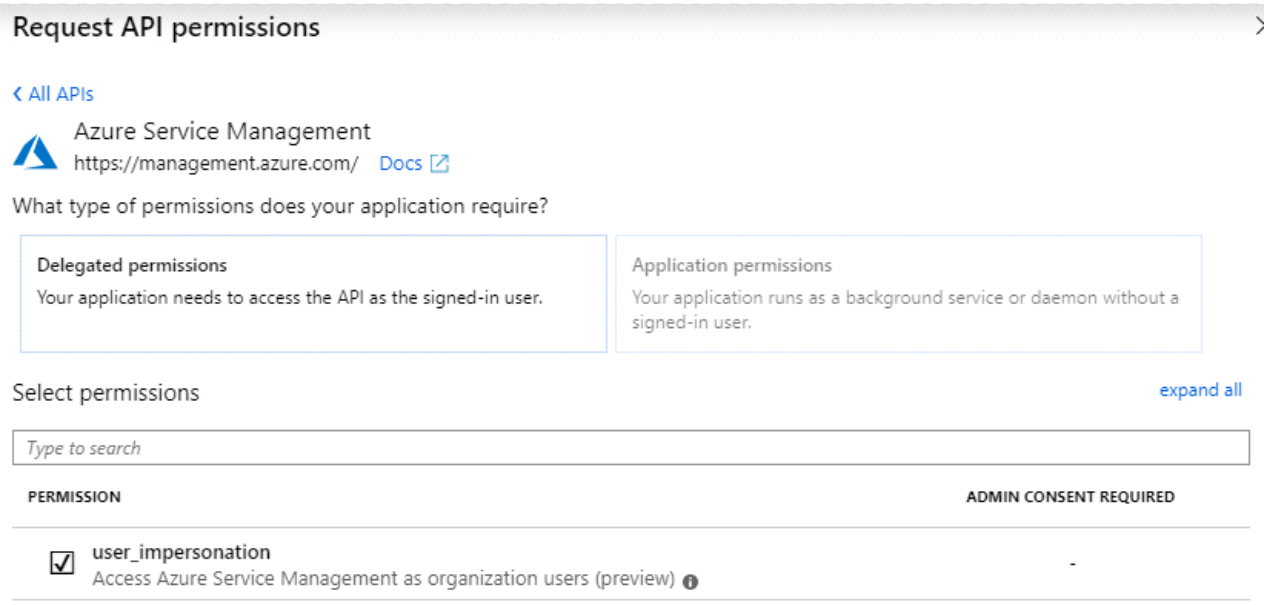
Windows Azure サービス管理 **API** 権限を追加する

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. **API** 権限 **>** 権限の追加 を選択します。

3. **Microsoft API** の下で、**Azure Service Management** を選択します。

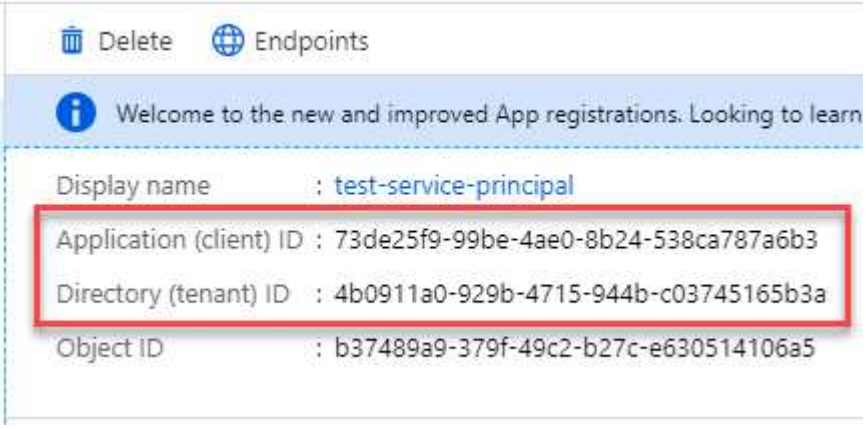


4. 組織ユーザーとして **Azure** サービス管理にアクセスする を選択し、権限の追加 を選択します。



アプリケーションのアプリケーション**ID**とディレクトリ**ID**を取得します

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. アプリケーション **(**クライアント**) ID** と ディレクトリ **(**テナント**) ID** をコピーします。



Azure アカウントをコンソールに追加するときは、アプリケーションのアプリケーション (クライアント) ID とディレクトリ (テナント) ID を指定する必要があります。コンソールは ID を使用してプログラムでサインインします。

クライアントシークレットを作成する

1. **Microsoft Entra ID** サービスを開きます。
2. *アプリ登録*を選択し、アプリケーションを選択します。
3. *証明書とシークレット > 新しいクライアント シークレット*を選択します。
4. シークレットの説明と期間を指定します。
5. *追加*を選択します。
6. クライアント シークレットの値をコピーします。

結果

これでサービス プリンシパルが設定され、アプリケーション (クライアント) ID、ディレクトリ (テナント) ID、およびクライアント シークレットの値がコピーされているはずです。 Azure アカウントを追加するときに、コンソールにこの情報を入力する必要があります。

ステップ**5:** コンソールエージェントをインストールする

前提条件が完了したら、独自の Linux ホストにソフトウェアを手動でインストールできます。

開始する前に

次のものが必要です:

- コンソール エージェントをインストールするためのルート権限。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。

  インストール後にプロキシ サーバーを構成するオプションがありますが、これを行うにはコンソール エージェントを再起動する必要があります。

- プロキシ サーバーが HTTPS を使用する場合、またはプロキシがインターセプト プロキシである場合は、CA 署名証明書。

> コンソール エージェントを手動でインストールする場合、透過プロキシ サーバーの証明書を設定することはできません。透過プロキシ サーバーの証明書を設定する必要がある場合は、インストール後にメンテナンス コンソールを使用する必要があります。詳細はこちら "エージェントメンテナンスコンソール"。

- カスタム ロールを通じて必要な Azure アクセス許可を提供できるように、Azure の VM で有効になっているマネージド ID。

  "Microsoft Azure ドキュメント: Azure ポータルを使用して VM 上の Azure リソースのマネージド ID を構成する"

タスク概要

インストール後、新しいバージョンが利用可能な場合、コンソール エージェントは自動的に更新されます。

手順

1. ホストに *http_proxy* または *https_proxy* システム変数が設定されている場合は、それらを削除します。

```
unset http_proxy
unset https_proxy
```

これらのシステム変数を削除しないと、インストールは失敗します。

2. コンソール エージェント ソフトウェアをダウンロードし、Linux ホストにコピーします。 NetApp ConsoleまたはNetAppサポート サイトからダウンロードできます。
   - NetApp Console: エージェント > 管理 > エージェントのデプロイ > オンプレミス > 手動インストールに移動します。

     エージェント インストーラー ファイルのダウンロードまたはファイルへの URL を選択します。

   - NetAppサポート サイト (コンソールにまだアクセスできない場合に必要) "NetAppサポート サイト" 、
3. スクリプトを実行するための権限を割り当てます。

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

<version> は、ダウンロードしたコンソール エージェントのバージョンです。

4. Government Cloud 環境にインストールする場合は、構成チェックを無効にします。"手動インストールの構成チェックを無効にする方法を説明します。"
5. インストール スクリプトを実行します。

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy
server> --cacert <path and file name of a CA-signed certificate>
```

ネットワークでインターネットアクセスにプロキシが必要な場合は、プロキシ情報を追加する必要があります。インストール中に明示的にプロキシを追加できます。 `--proxy`および `--cacert`パラメータはオプションであり、追加を要求されることはありません。明示的なプロキシ サーバがある場合は、示されているようにパラメータを入力する必要があります。

透過プロキシを設定する場合は、インストール後に設定できます。 "エージェントメンテナンスコンソールについて学ぶ"

+

CA 署名証明書を使用して明示的なプロキシ サーバーを構成する例を次に示します。

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+
`--proxy` 次のいずれかの形式を使用して、Console エージェントが HTTP または HTTPS プロキシ サーバを使用するように設定します：

+ * http://address:port * http://user-name:password@address:port * http://domain-name%92user-name:password@address:port * https://address:port * https://user-name:password@address:port * https://domain-name%92user-name:password@address:port

+ 以下の点に注意してください：

+ ユーザーは、ローカル ユーザーまたはドメイン ユーザーにすることができます。 ドメイン ユーザーの場合は、上記のように \ の ASCII コードを使用する必要があります。 **Console** エージェントは、**@** 文字を含むユーザー名またはパスワードをサポートしていません。 パスワードに次の特殊文字が含まれている場合は、その特殊文字の前にバックスラッシュを付けてエスケープする必要があります：& または !

+ 例：

+ http://bxpproxyuser:netapp1\!@address:3128

1. Podman を使用した場合は、aardvark-dns ポートを調整する必要があります。

   a. コンソール エージェント仮想マシンに SSH で接続します。

   b. podman */usr/share/containers/containers.conf* ファイルを開き、Aardvark DNS サービス用に選択したポートを変更します。たとえば、54 に変更します。

   ```
   vi /usr/share/containers/containers.conf
   ```

   例えば：

   ```
   # Port to use for dns forwarding daemon with netavark in rootful bridge
   # mode and dns enabled.
   # Using an alternate port might be useful if other DNS services should
   # run on the machine.
   #
   dns_bind_port = 54
   ```

   a. コンソール エージェント仮想マシンを再起動します。

2. インストールが完了するまでお待ちください。

   プロキシ サーバーを指定した場合、インストールの最後に、コンソール エージェント サービス (occm) が 2 回再起動します。

インストールが失敗した場合は、インストール レポートとログを表示して問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

1. コンソール エージェント仮想マシンに接続しているホストから Web ブラウザを開き、次の URL を入力します。

https://ipaddress

2. ログイン後、コンソール エージェントを設定します。
   a. コンソール エージェントに関連付ける組織を指定します。
   b. システムの名前を入力します。
   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      以下の手順ではコンソールを標準モードで使用する方法について説明しているため、制限モードは無効にしておく必要があります。安全な環境があり、このアカウントをバックエンド サービスから切断する場合にのみ、制限モードを有効にする必要があります。もしそうなら、"NetApp Consoleを制限モードで使い始めるための手順に従います" 。

   d. *始めましょう*を選択します。

コンソール エージェントを作成したのと同じ Azure サブスクリプションに Azure Blob ストレージがある場合は、システム ページに Azure Blob ストレージ システムが自動的に表示されます。 "NetApp ConsoleからAzure Blob ストレージを管理する方法を学びます"

ステップ**6: NetApp Console**に権限を付与する

コンソール エージェントをインストールしたので、以前に設定した Azure アクセス許可をコンソール エージェントに付与する必要があります。権限を付与すると、コンソールで Azure のデータとストレージ インフラストラクチャを管理できるようになります。

カスタムロール

Azure ポータルに移動し、1 つ以上のサブスクリプションのコンソール エージェント仮想マシンに Azure カスタム ロールを割り当てます。

手順

1. Azure ポータルから サブスクリプション サービスを開き、サブスクリプションを選択します。

   サブスクリプション サービスからロールを割り当てることが重要です。これは、サブスクリプション レベルでのロール割り当ての範囲を指定するためです。 *scope* は、アクセスが適用されるリソースのセットを定義します。別のレベル (たとえば、仮想マシン レベル) でスコープを指定すると、NetApp Console内からアクションを完了する機能に影響します。

   "Microsoft Azure ドキュメント: Azure RBAC のスコープを理解する"

2. アクセス制御 **(IAM)** > 追加 > ロール割り当ての追加 を選択します。
3. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。

   コンソール オペレーターは、ポリシーで提供されるデフォルト名です。ロールに別の名前を選択した場合は、代わりにその名前を選択します。

4. *メンバー*タブで、次の手順を実行します。
   a. マネージド **ID** へのアクセスを割り当てます。
   b. *メンバーの選択*を選択し、コンソール エージェント仮想マシンが作成されたサブスクリプションを選択し、*マネージド ID*の下で*仮想マシン*を選択して、コンソール エージェント仮想マシンを選択します。
   c. *選択*を選択します。
   d. *次へ*を選択します。
   e. *レビュー + 割り当て*を選択します。
   f. 追加の Azure サブスクリプションのリソースを管理する場合は、そのサブスクリプションに切り替えて、これらの手順を繰り返します。

次の手順

に行く "NetApp Console"コンソール エージェントの使用を開始します。

サービスプリンシパル

手順

1. *管理 > 資格情報*を選択します。
2. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: **Microsoft Azure >** エージェント を選択します。
   b. 資格情報の定義: 必要な権限を付与する Microsoft Entra サービス プリンシパルに関する情報を入力します。
      - アプリケーション（クライアント）ID
      - ディレクトリ（テナント）ID

- クライアントシークレット

c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。

d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

結果

これで、コンソール エージェントには、ユーザーに代わって Azure でアクションを実行するために必要なアクセス許可が付与されました。

Google Cloud

Google Cloud のコンソール エージェントのインストール オプション

Google Cloud でコンソール エージェントを作成するには、いくつかの方法があります。 NetApp Consoleから直接行うのが最も一般的な方法です。

次のインストール オプションが利用可能です。

- "コンソールから直接コンソールエージェントを作成する"（これが標準オプションです）

  このアクションにより、選択した VPC で Linux とコンソール エージェント ソフトウェアを実行する VM インスタンスが起動します。

- "Google プラットフォームを使用してコンソール エージェントを作成する"

  このアクションでは、Linux とコンソール エージェント ソフトウェアを実行する VM インスタンスも起動しますが、デプロイはコンソールからではなく Google Cloud から直接開始されます。

- "自分のLinuxホストにソフトウェアをダウンロードして手動でインストールする"

選択したインストール オプションは、インストールの準備方法に影響します。これには、Google Cloud でリソースを認証および管理するために必要な権限をコンソールに付与する方法が含まれます。

NetApp Consoleから Google Cloud にコンソール エージェントを作成する

コンソールから Google Cloud にコンソール エージェントを作成できます。ネットワークを設定し、Google Cloud の権限を準備し、Google Cloud API を有効にして、コンソール エージェントを作成する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ネットワークを設定する

コンソール エージェントがターゲット ネットワークへの接続とアウトバウンド インターネット アクセスを使用してリソースを管理できるように、ネットワークを設定します。

VPCとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VPC とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	Google Cloud 内のリソースを管理します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェア アップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。

エンドポイント	目的
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォール ルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

NetAppコンソールから接続されたエンドポイント

SaaS レイヤーを通じて提供される Web ベースのNetApp Consoleを使用すると、複数のエンドポイントに接続してデータ管理タスクが完了します。これには、コンソールからコンソール エージェントを展開するために接続されるエンドポイントが含まれます。

"NetAppコンソールから接続されたエンドポイントのリストを表示します" 。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、このネットワーク要件を実装します。

ステップ**2:** コンソールエージェントを作成するための権限を設定する

コンソールからコンソール エージェントをデプロイする前に、コンソール エージェント VM をデプロイする Google プラットフォーム ユーザーの権限を設定する必要があります。

手順

1. Google プラットフォームでカスタムロールを作成します。

   a. 次の権限を含む YAML ファイルを作成します。

```
title: Console agent deployment policy
description: Permissions for the user who deploys the Console agent
stage: GA
includedPermissions:

- cloudbuild.builds.get
- compute.disks.create
- compute.disks.get
- compute.disks.list
- compute.disks.setLabels
- compute.disks.use
- compute.firewalls.create
- compute.firewalls.delete
- compute.firewalls.get
- compute.firewalls.list
- compute.globalOperations.get
- compute.images.get
- compute.images.getFromFamily
- compute.images.list
```

```
- compute.images.useReadOnly
- compute.instances.attachDisk
- compute.instances.create
- compute.instances.get
- compute.instances.list
- compute.instances.setDeletionProtection
- compute.instances.setLabels
- compute.instances.setMachineType
- compute.instances.setMetadata
- compute.instances.setTags
- compute.instances.start
- compute.instances.updateDisplayDevice
- compute.machineTypes.get
- compute.networks.get
- compute.networks.list
- compute.networks.updatePolicy
- compute.projects.get
- compute.regions.get
- compute.regions.list
- compute.subnetworks.get
- compute.subnetworks.list
- compute.zoneOperations.get
- compute.zones.get
- compute.zones.list
- config.deployments.create
- config.operations.get
- config.deployments.delete
- config.deployments.deleteState
- config.deployments.get
- config.deployments.getState
- config.deployments.list
- config.deployments.update
- config.deployments.updateState
- config.previews.get
- config.previews.list
- config.revisions.get
- config.resources.list
- deploymentmanager.compositeTypes.get
- deploymentmanager.compositeTypes.list
- deploymentmanager.deployments.create
- deploymentmanager.deployments.delete
- deploymentmanager.deployments.get
- deploymentmanager.deployments.list
- deploymentmanager.manifests.get
- deploymentmanager.manifests.list
- deploymentmanager.operations.get
```

```
- deploymentmanager.operations.list
- deploymentmanager.resources.get
- deploymentmanager.resources.list
- deploymentmanager.typeProviders.get
- deploymentmanager.typeProviders.list
- deploymentmanager.types.get
- deploymentmanager.types.list
- resourcemanager.projects.get
- compute.instances.setServiceAccount
- iam.serviceAccounts.actAs
- iam.serviceAccounts.create
- iam.serviceAccounts.list
- iam.serviceAccountKeys.create
- storage.buckets.create
- storage.buckets.get
- storage.objects.create
- storage.folders.create
- storage.objects.list
```

b. Google Cloud から Cloud Shell を有効にします。

c. 必要な権限を含む YAML ファイルをアップロードします。

d. カスタムロールを作成するには、 `gcloud iam roles create`指示。

次の例では、プロジェクト レベルで「agentDeployment」という名前のロールを作成します。

gcloud iam ロール コネクタデプロイメントを作成 --project=myproject --file=agent-deployment.yaml

"Google Cloud ドキュメント: カスタムロールの作成と管理"

2. このカスタムロールは、コンソールから、または gcloud を使用してコンソール エージェントをデプロイするユーザーに割り当てます。

"Google Cloud ドキュメント: 単一のロールを付与する"

ステップ **3:** エージェントで使用する **Google Cloud** サービス アカウントを作成する

Google Cloud 内のリソースを管理するためにコンソールが必要とする権限をコンソール エージェントに付与するには、Google Cloud サービス アカウントが必要です。コンソール エージェントを作成するときは、このサービス アカウントをコンソール エージェント VM に関連付ける必要があります。

以降のリリースで新しい権限が追加された場合、カスタム ロールを更新するのはお客様の責任となります。新しい権限が必要な場合は、リリース ノートに記載されます。

手順

1. Google Cloud でカスタムロールを作成します。

a. 以下の内容を含むYAMLファイルを作成します。"コンソールエージェントのサービスアカウント権限"。

b. Google Cloud から Cloud Shell を有効にします。

c. 必要な権限を含む YAML ファイルをアップロードします。

d. カスタムロールを作成するには、 `gcloud iam roles create`指示。

次の例では、プロジェクト レベルで「agent」という名前のロールを作成します。

```
gcloud iam roles create connector --project=myproject --file=agent.yaml
```

"Google Cloud ドキュメント: カスタムロールの作成と管理"

2. Google Cloud でサービス アカウントを作成し、そのサービス アカウントにロールを割り当てます。

a. IAM & Admin サービスから、サービス アカウント **>** サービス アカウントの作成 を選択します。

b. サービス アカウントの詳細を入力し、[作成して続行] を選択します。

c. 作成したロールを選択します。

d. 残りの手順を完了してロールを作成します。

"Google Cloud ドキュメント: サービス アカウントの作成"

3. コンソール エージェントが存在するプロジェクトとは異なるプロジェクトにCloud Volumes ONTAPシステムを展開する予定の場合は、コンソール エージェントのサービス アカウントにそれらのプロジェクトへのアクセス権を付与する必要があります。

たとえば、コンソール エージェントがプロジェクト 1 にあり、プロジェクト 2 にCloud Volumes ONTAPシステムを作成するとします。プロジェクト 2 のサービス アカウントにアクセス権を付与する必要があります。

a. IAM & Admin サービスから、 Cloud Volumes ONTAPシステムを作成する Google Cloud プロジェクトを選択します。

b. **IAM** ページで、アクセスを許可 を選択し、必要な詳細を入力します。

- コンソール エージェントのサービス アカウントの電子メールを入力します。
- コンソール エージェントのカスタム ロールを選択します。
- *保存*を選択します。

詳細については、 "Google Cloud ドキュメント"

ステップ**4:** 共有**VPC**権限を設定する

共有 VPC を使用してリソースをサービス プロジェクトにデプロイする場合は、権限を準備する必要があります。

この表は参考用であり、IAM 構成が完了すると、環境に権限表が反映されるはずです。

共有 **VPC** 権限を表示する

身元	クリエイター	開催地	サービスプロジェクトの権限	ホストプロジェクトの権限	目的
エージェントを展開するためのGoogleアカウント	カスタム	奉仕プロジェクト	"エージェント展開ポリシー"	compute.networkUser	サービスプロジェクトにエージェントをデプロイする
エージェントサービスアカウント	カスタム	奉仕プロジェクト	"エージェント サービス アカウント ポリシー"	compute.networkUser デプロイメントマネージャー.エディター	サービス プロジェクトでCloud Volumes ONTAPとサービスをデプロイおよび保守する
Cloud Volumes ONTAP サービスアカウント	カスタム	奉仕プロジェクト	storage.admin メンバー: serviceAccount.user としてのNetApp Consoleサービスアカウント	該当なし	（オプション） NetApp Cloud TieringおよびNetApp Backup and Recoveryの場合
Google API サービス エージェント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わってGoogle Cloud API と対話します。コンソールが共有ネットワークを使用できるようにします。
Google Compute Engine のデフォルトのサービスアカウント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わって、Google Cloud インスタンスとコンピューティング インフラストラクチャをデプロイします。コンソールが共有ネットワークを使用できるようにします。

注：

1. ファイアウォール ルールをデプロイメントに渡さず、コンソールで自動的に作成するように選択した場合にのみ、ホスト プロジェクトで deploymentmanager.editor が必要になります。ルールが指定されていない場合、 NetApp Consoleは、VPC0 ファイアウォール ルールを含むデプロイメントをホスト プロジェクトに作成します。
2. firewall.create と firewall.delete は、デプロイメントにファイアウォール ルールを渡さず、コンソールで自動的に作成するように選択した場合にのみ必要です。これらの権限は、コンソール アカウントの .yaml ファイルにあります。共有 VPC を使用して HA ペアを展開する場合、これらの権限は VPC1、2、3 のファイアウォール ルールを作成するために使用されます。他のすべてのデプロイメントでは、これらの権限は VPC0 のルールの作成にも使用されます。
3. クラウド階層化の場合、階層化サービス アカウントには、プロジェクト レベルだけでなく、サービス アカウントに対する serviceAccount.user ロールが必要です。現在、プロジェクト レベルで serviceAccount.user を割り当てると、getIAMPolicy を使用してサービス アカウントをクエリしても

権限が表示されません。

ステップ5: Google Cloud APIを有効にする

Console エージェントとCloud Volumes ONTAPをデプロイする前に、いくつかの Google Cloud API を有効にする必要があります。

手順

1. プロジェクトで次の Google Cloud API を有効にします。
   - クラウド デプロイメント マネージャー V2 API
   - クラウド インフラストラクチャ マネージャー API
   - クラウドロギングAPI
   - クラウド リソース マネージャー API
   - コンピューティングエンジン API
   - アイデンティティとアクセス管理 (IAM) API
   - Cloud Key Management Service（KMS）API（お客様が管理する暗号化キー（CMEK）でNetApp Backup and Recoveryを使用する予定の場合のみ必要）
   - Cloud Quotas API（Infrastructure Managerを使用したCloud Volumes ONTAPデプロイメントに必要）

"Google Cloud ドキュメント: API の有効化"

ステップ6: コンソールエージェントを作成する

コンソールから直接コンソール エージェントを作成します。

コンソール エージェントを作成すると、デフォルト構成を使用して Google Cloud に仮想マシン インスタンスがデプロイされます。コンソール エージェントを作成した後、CPU や RAM が少ない小さな VM インスタンスに切り替えないでください。"コンソールエージェントのデフォルト構成について学習します" 。

Google Cloud にエージェントをデプロイすると、エージェントによってデプロイ ファイルを保存するためのバケットが作成されます。

開始する前に

次のものが必要です:

- コンソール エージェントとコンソール エージェント VM のサービス アカウントを作成するために必要な Google Cloud 権限。
- ネットワーク要件を満たす VPC とサブネット。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。

手順

1. *管理 > エージェント*を選択します。
2. *概要*ページで、*エージェントのデプロイ > Google Cloud*を選択します。

3. *エージェントのデプロイ*ページで、必要なものの詳細を確認します。次の 2 つのオプションがあります。
   a. 製品内ガイドを使用して展開の準備をするには、[続行] を選択します。製品内ガイドの各ステップには、ドキュメントのこのページに記載されている情報が含まれています。
   b. このページの手順に従ってすでに準備している場合は、[展開にスキップ] を選択します。
4. ウィザードの手順に従ってコンソール エージェントを作成します。
   - プロンプトが表示されたら、仮想マシン インスタンスを作成するために必要な権限を持つ Google アカウントにログインします。

     このフォームは Google が所有し、ホストしています。資格情報がNetAppに提供されていません。

   - 詳細: 仮想マシン インスタンスの名前を入力し、タグを指定して、プロジェクトを選択し、必要な権限を持つサービス アカウントを選択します (詳細については上記のセクションを参照してください)。
   - 場所: インスタンスのリージョン、ゾーン、VPC、サブネットを指定します。
   - ネットワーク: パブリック IP アドレスを有効にするかどうかを選択し、オプションでプロキシ構成を指定します。
   - ネットワーク タグ: 透過プロキシを使用している場合は、コンソール エージェント インスタンスにネットワーク タグを追加します。ネットワーク タグは小文字で始まる必要があり、小文字、数字、ハイフンを含めることができます。タグは小文字または数字で終わる必要があります。たとえば、「console-agent-proxy」というタグを使用できます。
   - ファイアウォール ポリシー: 新しいファイアウォール ポリシーを作成するか、必要な受信ルールと送信ルールを許可する既存のファイアウォール ポリシーを選択するかを選択します。

     "Google Cloud のファイアウォール ルール"

5. 選択内容を確認して、セットアップが正しいことを確認します。
   a. エージェント構成の検証 チェック ボックスはデフォルトでオンになっており、展開時にコンソールによってネットワーク接続要件が検証されます。コンソールがエージェントの展開に失敗した場合、トラブルシューティングに役立つレポートが提供されます。デプロイメントが成功した場合、レポートは提供されません。

   まだ使用している場合は"以前のエンドポイント"エージェントのアップグレードに使用すると、検証が失敗し、エラーが発生します。これを回避するには、チェックボックスをオフにして検証チェックをスキップします。

6. *追加*を選択します。

   エージェントは約 10 分で準備完了します。プロセスが完了するまでページに留まってください。

結果

プロセスが完了すると、コンソール エージェントが使用できるようになります。

デプロイメントが失敗した場合は、コンソールからレポートとログをダウンロードして、問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

コンソール エージェントを作成したのと同じ Google Cloud アカウントに Google Cloud Storage バケットがある場合は、[システム] ページに Google Cloud Storage システムが自動的に表示されます。 "コンソールから Google Cloud Storage を管理する方法を学びます"

Google Cloud からコンソール エージェントを作成する

Google Cloud を使用して Google Cloud にコンソール エージェントを作成するには、ネットワークを設定し、Google Cloud の権限を準備し、Google Cloud API を有効にして、コンソール エージェントを作成する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ネットワークを設定する

コンソール エージェントがリソースを管理し、ターゲット ネットワークおよびインターネットに接続できるようにネットワークを設定します。

VPCとサブネット

コンソール エージェントを作成するときは、そのエージェントが存在する VPC とサブネットを指定する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	Google Cloud 内のリソースを管理します。

エンドポイント	目的
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

NetAppコンソールから接続されたエンドポイント

SaaS レイヤーを通じて提供される Web ベースのNetApp Consoleを使用すると、複数のエンドポイントに接続してデータ管理タスクが完了します。これには、コンソールからコンソール エージェントを展開するために接続されるエンドポイントが含まれます。

"NetAppコンソールから接続されたエンドポイントのリストを表示します" 。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

コンソール エージェントを作成した後、このネットワーク要件を実装します。

ステップ**2:** コンソールエージェントを作成するための権限を設定する

Google Cloud ユーザーが Google Cloud からコンソール エージェント VM をデプロイするための権限を設定します。

手順

1. Google プラットフォームでカスタムロールを作成します。

   a. 次の権限を含む YAML ファイルを作成します。

```
title: Console agent deployment policy
description: Permissions for the user who deploys the Console
agent
stage: GA
includedPermissions:

- cloudbuild.builds.get
- compute.disks.create
- compute.disks.get
- compute.disks.list
- compute.disks.setLabels
- compute.disks.use
- compute.firewalls.create
- compute.firewalls.delete
- compute.firewalls.get
- compute.firewalls.list
- compute.globalOperations.get
- compute.images.get
- compute.images.getFromFamily
- compute.images.list
- compute.images.useReadOnly
- compute.instances.attachDisk
- compute.instances.create
- compute.instances.get
- compute.instances.list
- compute.instances.setDeletionProtection
- compute.instances.setLabels
- compute.instances.setMachineType
- compute.instances.setMetadata
- compute.instances.setTags
- compute.instances.start
- compute.instances.updateDisplayDevice
- compute.machineTypes.get
- compute.networks.get
- compute.networks.list
- compute.networks.updatePolicy
- compute.projects.get
- compute.regions.get
- compute.regions.list
- compute.subnetworks.get
- compute.subnetworks.list
- compute.zoneOperations.get
- compute.zones.get
- compute.zones.list
- config.deployments.create
```

```
- config.operations.get
- config.deployments.delete
- config.deployments.deleteState
- config.deployments.get
- config.deployments.getState
- config.deployments.list
- config.deployments.update
- config.deployments.updateState
- config.previews.get
- config.previews.list
- config.revisions.get
- config.resources.list
- deploymentmanager.compositeTypes.get
- deploymentmanager.compositeTypes.list
- deploymentmanager.deployments.create
- deploymentmanager.deployments.delete
- deploymentmanager.deployments.get
- deploymentmanager.deployments.list
- deploymentmanager.manifests.get
- deploymentmanager.manifests.list
- deploymentmanager.operations.get
- deploymentmanager.operations.list
- deploymentmanager.resources.get
- deploymentmanager.resources.list
- deploymentmanager.typeProviders.get
- deploymentmanager.typeProviders.list
- deploymentmanager.types.get
- deploymentmanager.types.list
- resourcemanager.projects.get
- compute.instances.setServiceAccount
- iam.serviceAccounts.actAs
- iam.serviceAccounts.create
- iam.serviceAccounts.list
- iam.serviceAccountKeys.create
- storage.buckets.create
- storage.buckets.get
- storage.objects.create
- storage.folders.create
- storage.objects.list
```

b. Google Cloud から Cloud Shell を有効にします。

c. 必要な権限を含む YAML ファイルをアップロードします。

d. カスタムロールを作成するには、 `gcloud iam roles create`指示。

次の例では、プロジェクト レベルで「connectorDeployment」という名前のロールを作成します。

gcloud iam ロール コネクタデプロイメントを作成 --project=myproject --file=connector
-deployment.yaml

"Google Cloud ドキュメント: カスタムロールの作成と管理"

2. このカスタムロールを、Google Cloud からコンソール エージェントをデプロイするユーザーに割り当てます。

"Google Cloud ドキュメント: 単一のロールを付与する"

ステップ**3:** コンソールエージェント操作の権限を設定する

Google Cloud 内のリソースを管理するためにコンソールが必要とする権限をコンソール エージェントに付与するには、Google Cloud サービス アカウントが必要です。コンソール エージェントを作成するときは、このサービス アカウントをコンソール エージェント VM に関連付ける必要があります。

以降のリリースで新しい権限が追加された場合、カスタム ロールを更新するのはお客様の責任となります。新しい権限が必要な場合は、リリース ノートに記載されます。

手順

1. Google Cloud でカスタムロールを作成します。

   a. 以下の内容を含むYAMLファイルを作成します。"コンソールエージェントのサービスアカウント権限"。
   b. Google Cloud から Cloud Shell を有効にします。
   c. 必要な権限を含む YAML ファイルをアップロードします。
   d. カスタムロールを作成するには、 `gcloud iam roles create`指示。

      次の例では、プロジェクト レベルで「agent」という名前のロールを作成します。

      ```
      gcloud iam roles create connector --project=myproject --file=agent.yaml
      ```

   "Google Cloud ドキュメント: カスタムロールの作成と管理"

2. Google Cloud でサービス アカウントを作成し、そのサービス アカウントにロールを割り当てます。

   a. IAM & Admin サービスから、サービス アカウント **>** サービス アカウントの作成 を選択します。
   b. サービス アカウントの詳細を入力し、[作成して続行] を選択します。
   c. 作成したロールを選択します。
   d. 残りの手順を完了してロールを作成します。

      "Google Cloud ドキュメント: サービス アカウントの作成"

3. コンソール エージェントが存在するプロジェクトとは異なるプロジェクトにCloud Volumes ONTAPシステムを展開する予定の場合は、コンソール エージェントのサービス アカウントにそれらのプロジェクトへのアクセス権を付与する必要があります。

   たとえば、コンソール エージェントがプロジェクト 1 にあり、プロジェクト 2 にCloud Volumes ONTAP

システムを作成するとします。プロジェクト 2 のサービス アカウントにアクセス権を付与する必要があります。

a. IAM & Admin サービスから、 Cloud Volumes ONTAPシステムを作成する Google Cloud プロジェクトを選択します。

b. **IAM** ページで、アクセスを許可 を選択し、必要な詳細を入力します。

- コンソール エージェントのサービス アカウントの電子メールを入力します。
- コンソール エージェントのカスタム ロールを選択します。
- *保存*を選択します。

詳細については、 "Google Cloud ドキュメント"

ステップ**4:** 共有**VPC**権限を設定する

共有 VPC を使用してリソースをサービス プロジェクトにデプロイする場合は、権限を準備する必要があります。

この表は参考用であり、IAM 構成が完了すると、環境に権限表が反映されるはずです。

共有 **VPC** 権限を表示する

身元	クリエイター	開催地	サービスプロジェクトの権限	ホストプロジェクトの権限	目的
エージェントを展開するためのGoogleアカウント	カスタム	奉仕プロジェクト	"エージェント展開ポリシー"	compute.networkUser	サービスプロジェクトにエージェントをデプロイする
エージェントサービスアカウント	カスタム	奉仕プロジェクト	"エージェント サービス アカウント ポリシー"	compute.networkUser デプロイメントマネージャー.エディター	サービス プロジェクトでCloud Volumes ONTAPとサービスをデプロイおよび保守する
Cloud Volumes ONTAP サービスアカウント	カスタム	奉仕プロジェクト	storage.admin メンバー: serviceAccount.user としてのNetApp Consoleサービスアカウント	該当なし	（オプション） NetApp Cloud TieringおよびNetApp Backup and Recoveryの場合
Google API サービス エージェント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わって Google Cloud API と対話します。コンソールが共有ネットワークを使用できるようにします。
Google Compute Engine のデフォルトのサービスアカウント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わって、Google Cloud インスタンスとコンピューティング インフラストラクチャをデプロイします。コンソールが共有ネットワークを使用できるようにします。

注：

1. ファイアウォール ルールをデプロイメントに渡さず、コンソールで自動的に作成するように選択した場合にのみ、ホスト プロジェクトで deploymentmanager.editor が必要になります。ルールが指定されていない場合、 NetApp Consoleは、VPC0 ファイアウォール ルールを含むデプロイメントをホスト プロジェクトに作成します。
2. firewall.create と firewall.delete は、デプロイメントにファイアウォール ルールを渡さず、コンソールで自動的に作成するように選択した場合にのみ必要です。これらの権限は、コンソール アカウントの .yaml ファイルにあります。共有 VPC を使用して HA ペアを展開する場合、これらの権限は VPC1、2、3 のファイアウォール ルールを作成するために使用されます。他のすべてのデプロイメントでは、これらの権限は VPC0 のルールの作成にも使用されます。
3. クラウド階層化の場合、階層化サービス アカウントには、プロジェクト レベルだけでなく、サービス アカウントに対する serviceAccount.user ロールが必要です。現在、プロジェクト レベルで serviceAccount.user を割り当てると、getIAMPolicy を使用してサービス アカウントをクエリしても

権限が表示されません。

ステップ**5: Google Cloud API**を有効にする

Console エージェントとCloud Volumes ONTAP をデプロイする前に、いくつかの Google Cloud API を有効にします。

手順

1. プロジェクトで次の Google Cloud API を有効にします。
   - クラウド デプロイメント マネージャー V2 API
   - クラウド インフラストラクチャ マネージャー API
   - クラウドロギングAPI
   - クラウド リソース マネージャー API
   - コンピューティングエンジン API
   - アイデンティティとアクセス管理 (IAM) API
   - Cloud Key Management Service（KMS）API（お客様が管理する暗号化キー（CMEK）でNetApp Backup and Recoveryを使用する予定の場合のみ必要）
   - Cloud Quotas API（Infrastructure Managerを使用したCloud Volumes ONTAPデプロイメントに必要）

"Google Cloud ドキュメント: API の有効化"

ステップ**6:** コンソールエージェントを作成する

Google Cloud を使用してコンソール エージェントを作成します。

コンソール エージェントを作成すると、デフォルト構成で Google Cloud に VM インスタンスがデプロイされます。コンソール エージェントを作成した後、CPU や RAM が少ない小さな VM インスタンスに切り替えないでください。"コンソールエージェントのデフォルト構成について学習します" 。

開始する前に

次のものが必要です:

- コンソール エージェントとコンソール エージェント VM のサービス アカウントを作成するために必要な Google Cloud 権限。
- ネットワーク要件を満たす VPC とサブネット。
- VM インスタンスの要件を理解していること。
  - **CPU**: 8 コアまたは 8 vCPU
  - **RAM**: 32 GB
  - マシンタイプ: n2-standard-8 を推奨します。

    Console エージェントは、Shielded VM 機能をサポートする OS を搭載した VM インスタンス上の Google Cloud でサポートされます。

手順

1. お好みの方法で Google Cloud SDK にログインします。

   この例では、gcloud SDK がインストールされたローカル シェルを使用していますが、Google Cloud Shell を使用することもできます。

   Google Cloud SDKの詳細については、"Google Cloud SDK ドキュメント ページ" 。

2. 上記のセクションで定義されている必要な権限を持つユーザーとしてログインしていることを確認します。

   ```
   gcloud auth list
   ```

   出力には次のように表示されます。* ユーザー アカウントは、ログインに使用するユーザー アカウントです。

   ```
   Credentialed Accounts
   ACTIVE  ACCOUNT
        some_user_account@domain.com
   *    desired_user_account@domain.com
   To set the active account, run:
    $ gcloud config set account `ACCOUNT`
   Updates are available for some Cloud SDK components. To install them,
   please run:
   $ gcloud components update
   ```

3. 実行 `gcloud compute instances create`指示：

   ```
   gcloud compute instances create <instance-name>
     --machine-type=n2-standard-8
     --image-project=netapp-cloudmanager
     --image-family=cloudmanager
     --scopes=cloud-platform
     --project=<project>
     --service-account=<service-account>
     --zone=<zone>
     --no-address
     --tags <network-tag>
     --network <network-path>
     --subnet <subnet-path>
     --boot-disk-kms-key <kms-key-path>
   ```

   インスタンス名
   : VM インスタンスの希望するインスタンス名。

プロジェクト
: (オプション) VM をデプロイするプロジェクト。

サービスアカウント
: 手順 2 の出力で指定されたサービス アカウント。

ゾーン
: VMを展開するゾーン

住所なし
: (オプション) 外部 IP アドレスは使用されません (トラフィックをパブリック インターネットにルーティングするには、クラウド NAT またはプロキシが必要です)

ネットワークタグ
: (オプション) ネットワーク タグ付けを追加して、タグを使用してファイアウォール ルールをコンソール エージェント インスタンスにリンクします。

ネットワークパス
: (オプション) コンソールエージェントをデプロイするネットワークの名前を追加します (共有 VPC の場合はフルパスが必要です)

サブネットパス
: (オプション) コンソールエージェントをデプロイするサブネットの名前を追加します (共有 VPC の場合はフルパスが必要です)

kmsキーパス
: (オプション) コンソール エージェントのディスクを暗号化するための KMS キーを追加します (IAM 権限も適用する必要があります)

  これらの旗の詳細については、"Google Cloud Compute SDK ドキュメント" 。

コマンドを実行すると、コンソール エージェントがデプロイされます。コンソール エージェント インスタンスとソフトウェアは、約 5 分以内に実行されるはずです。

4. Web ブラウザを開き、コンソール エージェント ホストの URL を入力します。

   コンソール ホスト URL は、ホストの構成に応じて、ローカルホスト、プライベート IP アドレス、またはパブリック IP アドレスになります。たとえば、コンソール エージェントがパブリック IP アドレスのないパブリック クラウドにある場合は、コンソール エージェント ホストに接続しているホストのプライベート IP アドレスを入力する必要があります。

5. ログイン後、コンソール エージェントを設定します。

   a. コンソール エージェントに関連付けるコンソール組織を指定します。

      "アイデンティティとアクセス管理について学ぶ" 。

   b. システムの名前を入力します。

結果

コンソール エージェントがインストールされ、コンソール組織に設定されました。

ウェブブラウザを開いて、 "NetApp Console"コンソール エージェントの使用を開始します。

Google Cloud にコンソール エージェントを手動でインストールする

独自の Linux ホストに Console エージェントを手動でインストールするには、ホストの要件を確認し、ネットワークを設定し、Google Cloud の権限を準備し、Google Cloud API を有効にし、Console をインストールして、準備した権限を付与する必要があります。

開始する前に

- あなたは"コンソールエージェントの理解"。
- 確認すべき"コンソールエージェントの制限"。

ステップ**1:** ホストの要件を確認する

コンソール エージェント ソフトウェアは、特定のオペレーティング システム要件、RAM 要件、ポート要件などを満たすホスト上で実行する必要があります。

> コンソール エージェントは、19000 ～ 19200 の UID と GID の範囲を予約します。この範囲は固定されており、変更することはできません。ホスト上のサードパーティ ソフトウェアがこの範囲内の UID または GID を使用している場合、エージェントのインストールは失敗します。NetApp、競合を回避するためにサードパーティ ソフトウェアがインストールされていないホストの使用を推奨しています。

専用ホスト

コンソール エージェントには専用のホストが必要です。次のサイズ要件を満たすアーキテクチャであれば、どれでもサポートされます。

- CPU: 8コアまたは8vCPU
- メモリ: 32 GB
- ディスク容量: ホストには 165 GB が推奨され、パーティション要件は次のとおりです。
  - `/opt`: 120 GiBの空き容量が必要です

    エージェントは `/opt`インストールするには `/opt/application/netapp`ディレクトリとその内容。
  - `/var`: 40 GiBの空き容量が必要です

    コンソールエージェントにはこのスペースが必要です `/var`Podman または Docker は、このディレクトリ内にコンテナを作成するように設計されているためです。具体的には、`/var/lib/containers/storage`ディレクトリと `/var/lib/docker`Docker用。このスペースでは外部マウントまたはシンボリックリンクは機能しません。

Google Cloud マシンタイプ

CPU と RAM の要件を満たすインスタンス タイプ。NetAppn2-standard-8 を推奨しています。

コンソールエージェントは、Google Cloud の VM インスタンスで、以下の OS がサポートする環境でサポートされます。 "シールドされたVMの機能"

ハイパーバイザー

サポートされているオペレーティング システムを実行することが認定されているベア メタルまたはホスト型ハイパーバイザーが必要です。

オペレーティングシステムとコンテナの要件

コンソールを標準モードまたは制限モードで使用する場合、コンソール エージェントは次のオペレーティング システムでサポートされます。エージェントをインストールする前に、コンテナ オーケストレーション ツールが必要です。

オペレーティング システム	サポートされる**OS** バージョン	サポートされているエージェントのバージョン	必要なコンテナツール	**SELinux**
Red Hat Enterprise Linux		9.6 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	4.0.0 以降、コンソールが標準モードまたは制限モード	Podman バージョン 5.4.0 と podman-compose 1.5.0。 Podman の構成要件を表示する 。

オペレーティングシステム	サポートされるOSバージョン	サポートされているエージェントのバージョン	必要なコンテナツール	SELinux
強制モードまたは許可モードでサポートされます		9.1～9.4 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.9.4 と podman-compose 1.5.0。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます		8.6～8.10 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.6.1 または 4.9.4 と podman-compose 1.0.6。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます	Ubuntu		24.04 LTS	3.9.45 以降、NetApp Consoleが標準モードまたは制限モード
Docker エンジン 23.06 から 28.0.0。	サポート対象外		22.04 LTS	3.9.50以降

Google Cloud マシンタイプ

CPU と RAM の要件を満たすインスタンス タイプ。NetAppn2-standard-8 を推奨しています。

コンソールエージェントは、Google Cloud の VM インスタンスで、以下の OS がサポートする環境でサポートされます。 "シールドされたVMの機能"

ステップ**2: Podman**または**Docker Engine**をインストールする

オペレーティング システムに応じて、エージェントをインストールする前に Podman または Docker Engine のいずれかが必要になります。

- Red Hat Enterprise Linux 8 および 9 には Podman が必要です。

  サポートされているPodmanのバージョンを表示する 。

- Ubuntu には Docker Engine が必要です。

  サポートされている Docker エンジンのバージョンを表示する 。

例 **3.** 手順

ポッドマン

Podman をインストールして設定するには、次の手順に従います。

- podman.socket サービスを有効にして起動します
- Python3をインストールする
- podman-compose パッケージ バージョン 1.0.6 をインストールします。
- podman-composeをPATH環境変数に追加する
- Red Hat Enterprise Linux を使用している場合は、Podman バージョンが CNI ではなく Netavark Aardvark DNS を使用していることを確認してください。

DNS ポートの競合を避けるために、エージェントをインストールした後、aardvark-dns ポート (デフォルト: 53) を調整します。指示に従ってポートを構成します。

手順

1. ホストに podman-docker パッケージがインストールされている場合は削除します。

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Podman をインストールします。

   Podman は、公式の Red Hat Enterprise Linux リポジトリから入手できます。

   a. Red Hat Enterprise Linux 9.6 の場合:

   ```
   sudo dnf install podman-5:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   b. Red Hat Enterprise Linux 9.1 から 9.4 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   c. Red Hat Enterprise Linux 8 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

<version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

3. podman.socket サービスを有効にして起動します。

```
sudo systemctl enable --now podman.socket
```

4. python3 をインストールします。

```
sudo dnf install python3
```

5. システムにまだインストールされていない場合は、EPEL リポジトリ パッケージをインストールします。

   podman-compose は、Extra Packages for Enterprise Linux (EPEL) リポジトリから入手できるため、この手順は必須です。

6. Red Hat Enterprise 9 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

   +

   a. podman-compose パッケージ 1.5.0 をインストールします。

```
sudo dnf install podman-compose-1.5.0
```

7. Red Hat Enterprise Linux 8 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

   b. podman-compose パッケージ 1.0.6 をインストールします。

```
sudo dnf install podman-compose-1.0.6
```

使用して `dnf install`コマンドは、PATH 環境変数に podman-compose を追加するための要件を満たしています。インストールコマンドは、すでにインストールされているpodman-composeを/usr/binに追加します。 `secure_path`ホスト上のオプション。

c. Red Hat Enterprise Linux 8 を使用している場合は、Podman バージョンが CNI ではなく Aardvark DNS を備えた NetAvark を使用していることを確認します。

i. 次のコマンドを実行して、networkBackend が CNI に設定されているかどうかを確認します。

```
podman info | grep networkBackend
```

ii. networkBackendが `CNI`、それを変更する必要があります `netavark`。

iii. インストール `netavark`そして `aardvark-dns`次のコマンドを使用します。

```
dnf install aardvark-dns netavark
```

iv. 開く `/etc/containers/containers.conf`ファイルを編集し、network_backend オプションを変更して、「cni」の代わりに「netavark」を使用します。

もし `/etc/containers/containers.conf`存在しない場合は、設定を変更してください `/usr/share/containers/containers.conf`。

v. podman を再起動します。

```
systemctl restart podman
```

vi. 次のコマンドを使用して、networkBackend が「netavark」に変更されていることを確認します。

```
podman info | grep networkBackend
```

Dockerエンジン

Docker のドキュメントに従って Docker Engine をインストールします。

手順

1. "Dockerからのインストール手順を見る"

   サポートされている Docker エンジン バージョンをインストールするには、手順に従ってください。最新バージョンはコンソールでサポートされていないため、インストールしないでください。

2. Docker が有効になっていて実行されていることを確認します。

```
sudo systemctl enable docker && sudo systemctl start docker
```

ステップ**3:** ネットワークを設定する

コンソール エージェントがハイブリッド クラウド環境内のリソースとプロセスを管理できるようにネットワークを設定します。たとえば、ターゲット ネットワークへの接続が利用可能であること、および発信インターネット アクセスが利用可能であることを確認する必要があります。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

Webベースの**NetApp Console**を使用する際にコンピュータから接続されるエンドポイント

Web ブラウザからコンソールにアクセスするコンピュータは、複数のエンドポイントに接続できる必要があります。コンソール エージェントを設定し、コンソールを日常的に使用するには、コンソールを使用する必要があります。

"NetAppコンソールのネットワークを準備する" 。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	Google Cloud 内のリソースを管理します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。

エンドポイント	目的
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス

- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

ステップ**4:** コンソールエージェントの権限を設定する

Google Cloud 内のリソースを管理するためにコンソールが必要とする権限をコンソール エージェントに付与するには、Google Cloud サービス アカウントが必要です。コンソール エージェントを作成するときは、このサービス アカウントをコンソール エージェント VM に関連付ける必要があります。

以降のリリースで新しい権限が追加された場合、カスタム ロールを更新するのはお客様の責任となります。新しい権限が必要な場合は、リリース ノートに記載されます。

手順

1. Google Cloud でカスタムロールを作成します。

   a. 以下の内容を含むYAMLファイルを作成します。"コンソールエージェントのサービスアカウント権限"。
   b. Google Cloud から Cloud Shell を有効にします。
   c. 必要な権限を含む YAML ファイルをアップロードします。
   d. カスタムロールを作成するには、 `gcloud iam roles create`指示。

      次の例では、プロジェクト レベルで「agent」という名前のロールを作成します。

      ```
      gcloud iam roles create connector --project=myproject --file=agent.yaml
      ```

   "Google Cloud ドキュメント: カスタムロールの作成と管理"

2. Google Cloud でサービス アカウントを作成し、そのサービス アカウントにロールを割り当てます。
   a. IAM & Admin サービスから、サービス アカウント > サービス アカウントの作成 を選択します。
   b. サービス アカウントの詳細を入力し、[作成して続行] を選択します。
   c. 作成したロールを選択します。
   d. 残りの手順を完了してロールを作成します。

      "Google Cloud ドキュメント: サービス アカウントの作成"

3. コンソール エージェントが存在するプロジェクトとは異なるプロジェクトにCloud Volumes ONTAPシステムを展開する予定の場合は、コンソール エージェントのサービス アカウントにそれらのプロジェクトへのアクセス権を付与する必要があります。

   たとえば、コンソール エージェントがプロジェクト 1 にあり、プロジェクト 2 にCloud Volumes ONTAPシステムを作成するとします。プロジェクト 2 のサービス アカウントにアクセス権を付与する必要があります。

   a. IAM & Admin サービスから、 Cloud Volumes ONTAPシステムを作成する Google Cloud プロジェクトを選択します。
   b. **IAM** ページで、アクセスを許可 を選択し、必要な詳細を入力します。
      - コンソール エージェントのサービス アカウントの電子メールを入力します。
      - コンソール エージェントのカスタム ロールを選択します。
      - *保存*を選択します。

   詳細については、 "Google Cloud ドキュメント"

ステップ**5:** 共有**VPC**権限を設定する

共有 VPC を使用してリソースをサービス プロジェクトにデプロイする場合は、権限を準備する必要があります。

この表は参考用であり、IAM 構成が完了すると、環境に権限表が反映されるはずです。

共有 **VPC** 権限を表示する

身元	クリエイター	開催地	サービスプロジェクトの権限	ホストプロジェクトの権限	目的
エージェントを展開するためのGoogleアカウント	カスタム	奉仕プロジェクト	"エージェント展開ポリシー"	compute.networkUser	サービスプロジェクトにエージェントをデプロイする
エージェントサービスアカウント	カスタム	奉仕プロジェクト	"エージェント サービス アカウント ポリシー"	compute.networkUser デプロイメントマネージャー.エディター	サービス プロジェクトでCloud Volumes ONTAPとサービスをデプロイおよび保守する
Cloud Volumes ONTAP サービスアカウント	カスタム	奉仕プロジェクト	storage.admin メンバー: serviceAccount.user としてのNetApp Consoleサービスアカウント	該当なし	（オプション） NetApp Cloud TieringおよびNetApp Backup and Recoveryの場合
Google API サービス エージェント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わってGoogle Cloud API と対話します。コンソールが共有ネットワークを使用できるようにします。
Google Compute Engine のデフォルトのサービスアカウント	Google Cloud	奉仕プロジェクト	(デフォルト) エディター	compute.networkUser	デプロイメントに代わって、Google Cloud インスタンスとコンピューティング インフラストラクチャをデプロイします。コンソールが共有ネットワークを使用できるようにします。

注：

1. ファイアウォール ルールをデプロイメントに渡さず、コンソールで自動的に作成するように選択した場合にのみ、ホスト プロジェクトで deploymentmanager.editor が必要になります。ルールが指定されていない場合、 NetApp Consoleは、VPC0 ファイアウォール ルールを含むデプロイメントをホスト プロジェクトに作成します。
2. firewall.create と firewall.delete は、デプロイメントにファイアウォール ルールを渡さず、コンソールで自動的に作成するように選択した場合にのみ必要です。これらの権限は、コンソール アカウントの .yaml ファイルにあります。共有 VPC を使用して HA ペアを展開する場合、これらの権限は VPC1、2、3 のファイアウォール ルールを作成するために使用されます。他のすべてのデプロイメントでは、これらの権限は VPC0 のルールの作成にも使用されます。
3. クラウド階層化の場合、階層化サービス アカウントには、プロジェクト レベルだけでなく、サービス アカウントに対する serviceAccount.user ロールが必要です。現在、プロジェクト レベルで serviceAccount.user を割り当てると、getIAMPolicy を使用してサービス アカウントをクエリしても

権限が表示されません。

ステップ6: Google Cloud APIを有効にする

Google Cloud にコンソール エージェントをデプロイする前に、いくつかの Google Cloud API を有効にする必要があります。

手順

1. プロジェクトで次の Google Cloud API を有効にします。
   - クラウド デプロイメント マネージャー V2 API
   - クラウド インフラストラクチャ マネージャー API
   - クラウドロギングAPI
   - クラウド リソース マネージャー API
   - コンピューティングエンジン API
   - アイデンティティとアクセス管理 (IAM) API
   - Cloud Key Management Service（KMS）API（お客様が管理する暗号化キー（CMEK）でNetApp Backup and Recoveryを使用する予定の場合のみ必要）
   - Cloud Quotas API（Infrastructure Managerを使用したCloud Volumes ONTAPデプロイメントに必要）

"Google Cloud ドキュメント: API の有効化"

ステップ7: コンソールエージェントをインストールする

前提条件が完了したら、独自の Linux ホストにソフトウェアを手動でインストールできます。

エージェントをデプロイすると、デプロイ ファイルを保存するための Google Cloud バケットも作成されます。

開始する前に

次のものが必要です:

- コンソール エージェントをインストールするためのルート権限。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。

  インストール後にプロキシ サーバーを構成するオプションがありますが、これを行うにはコンソール エージェントを再起動する必要があります。

- プロキシ サーバーが HTTPS を使用する場合、またはプロキシがインターセプト プロキシである場合は、CA 署名証明書。

> ⓘ コンソール エージェントを手動でインストールする場合、透過プロキシ サーバーの証明書を設定することはできません。透過プロキシ サーバーの証明書を設定する必要がある場合は、インストール後にメンテナンス コンソールを使用する必要があります。詳細はこちら "エージェントメンテナンスコンソール"。

タスク概要

インストール後、新しいバージョンが利用可能な場合、コンソール エージェントは自動的に更新されます。

手順

1. ホストに *http_proxy* または *https_proxy* システム変数が設定されている場合は、それらを削除します。

```
unset http_proxy
unset https_proxy
```

これらのシステム変数を削除しないと、インストールは失敗します。

2. コンソール エージェント ソフトウェアをダウンロードし、Linux ホストにコピーします。 NetApp ConsoleまたはNetAppサポート サイトからダウンロードできます。
   - NetApp Console: エージェント > 管理 > エージェントのデプロイ > オンプレミス > 手動インストールに移動します。

     エージェント インストーラー ファイルのダウンロードまたはファイルへの URL を選択します。

   - NetAppサポート サイト (コンソールにまだアクセスできない場合に必要) "NetAppサポート サイト" 、
3. スクリプトを実行するための権限を割り当てます。

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

<version> は、ダウンロードしたコンソール エージェントのバージョンです。

4. Government Cloud 環境にインストールする場合は、構成チェックを無効にします。"手動インストールの構成チェックを無効にする方法を説明します。"
5. インストール スクリプトを実行します。

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy
server> --cacert <path and file name of a CA-signed certificate>
```

ネットワークでインターネットアクセスにプロキシが必要な場合は、プロキシ情報を追加する必要があります。インストール中に明示的にプロキシを追加できます。 `--proxy`および `--cacert`パラメータはオプションであり、追加を要求されることはありません。明示的なプロキシ サーバがある場合は、示されているようにパラメータを入力する必要があります。

透過プロキシを設定する場合は、インストール後に設定できます。 "エージェントメンテナンスコンソールについて学ぶ"

+

CA 署名証明書を使用して明示的なプロキシ サーバーを構成する例を次に示します。

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+
`--proxy` 次のいずれかの形式を使用して、Console エージェントが HTTP または HTTPS プロキシ サーバを使用するように設定します：

+ * http://address:port * http://user-name:password@address:port * http://domain-name%92user-name:password@address:port * https://address:port * https://user-name:password@address:port * https://domain-name%92user-name:password@address:port

+ 以下の点に注意してください：

+ ユーザーは、ローカル ユーザーまたはドメイン ユーザーにすることができます。 ドメイン ユーザーの場合は、上記のように \ の ASCII コードを使用する必要があります。 **Console** エージェントは、**@** 文字を含むユーザー名またはパスワードをサポートしていません。 パスワードに次の特殊文字が含まれている場合は、その特殊文字の前にバックスラッシュを付けてエスケープする必要があります：& または !

+ 例：

+ http://bxpproxyuser:netapp1\!@address:3128

1. Podman を使用した場合は、aardvark-dns ポートを調整する必要があります。

   a. コンソール エージェント仮想マシンに SSH で接続します。

   b. podman */usr/share/containers/containers.conf* ファイルを開き、Aardvark DNS サービス用に選択したポートを変更します。たとえば、54 に変更します。

      ```
      vi /usr/share/containers/containers.conf
      ```

      例えば：

      ```
      # Port to use for dns forwarding daemon with netavark in rootful bridge
      # mode and dns enabled.
      # Using an alternate port might be useful if other DNS services should
      # run on the machine.
      #
      dns_bind_port = 54
      ```

   a. コンソール エージェント仮想マシンを再起動します。

2. インストールが完了するまでお待ちください。

   プロキシ サーバーを指定した場合、インストールの最後に、コンソール エージェント サービス (occm) が 2 回再起動します。

インストールが失敗した場合は、インストール レポートとログを表示して問題の解決に役立てることができます。"インストールの問題をトラブルシューティングする方法を学びます。"

1. コンソール エージェント仮想マシンに接続しているホストから Web ブラウザを開き、次の URL を入力します。

   https://ipaddress

2. ログイン後、コンソール エージェントを設定します。

   a. コンソール エージェントに関連付ける組織を指定します。

   b. システムの名前を入力します。

   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      以下の手順ではコンソールを標準モードで使用する方法について説明しているため、制限モードは無効にしておく必要があります。安全な環境があり、このアカウントをバックエンド サービスから切断する場合にのみ、制限モードを有効にする必要があります。もしそうなら、"NetApp Consoleを制限モードで使い始めるための手順に従います" 。

   d. *始めましょう*を選択します。

   インストールが失敗した場合は、トラブルシューティングに役立つログとレポートを表示できます。"インストールの問題をトラブルシューティングする方法を学びます。"

コンソール エージェントを作成したのと同じ Google Cloud アカウントに Google Cloud Storage バケットがある場合は、[システム] ページに Google Cloud Storage システムが自動的に表示されます。 "NetApp Consoleから Google Cloud Storage を管理する方法を学びます"

ステップ**8:** コンソールエージェントに権限を付与する

以前に設定した Google Cloud 権限をコンソール エージェントに提供する必要があります。権限を付与すると、コンソール エージェントが Google Cloud 内のデータとストレージ インフラストラクチャを管理できるようになります。

手順

1. Google Cloud ポータルに移動し、サービス アカウントをコンソール エージェント VM インスタンスに割り当てます。

   "Google Cloud ドキュメント: インスタンスのサービス アカウントとアクセス スコープの変更"

2. 他の Google Cloud プロジェクトのリソースを管理する場合は、コンソール エージェントのロールを持つサービス アカウントをそのプロジェクトに追加してアクセス権を付与します。プロジェクトごとにこの手順を繰り返す必要があります。

オンプレミスにエージェントをインストールする

オンプレミスにコンソールエージェントを手動でインストールする

オンプレミスにコンソール エージェントをインストールし、ログインして、コンソール

組織で動作するように設定します。

VMWare ユーザーの場合は、OVA を使用して VCenter にコンソール エージェントをインストールできます。"VCenter にエージェントをインストールする方法の詳細について説明します。"

インストールする前に、ホスト (VM または Linux ホスト) が要件を満たしていること、およびコンソール エージェントがインターネットと対象ネットワークへの送信アクセスできることを確認する必要があります。NetAppデータ サービス、またはCloud Volumes ONTAPなどのクラウド ストレージ オプションを使用する予定の場合は、コンソール エージェントがユーザーに代わってクラウド内でアクションを実行できるように、クラウド プロバイダーで資格情報を作成してコンソールに追加する必要があります。

コンソールエージェントのインストールの準備

コンソール エージェントをインストールする前に、インストール要件を満たすホスト マシンがあることを確認する必要があります。また、ネットワーク管理者と協力して、コンソール エージェントが必要なエンドポイントへの送信アクセスと対象ネットワークへの接続を持っていることを確認する必要があります。

コンソールエージェントホストの要件を確認する

オペレーティング システム、RAM、およびポートの要件を満たす x86 ホストでコンソール エージェントを実行します。コンソール エージェントをインストールする前に、ホストがこれらの要件を満たしていることを確認してください。

コンソール エージェントは、19000 ～ 19200 の UID と GID の範囲を予約します。この範囲は固定されており、変更することはできません。ホスト上のサードパーティ ソフトウェアがこの範囲内の UID または GID を使用している場合、エージェントのインストールは失敗します。NetApp、競合を回避するためにサードパーティ ソフトウェアがインストールされていないホストの使用を推奨しています。

専用ホスト

コンソール エージェントには専用のホストが必要です。次のサイズ要件を満たすアーキテクチャであれば、どれでもサポートされます。

- CPU: 8コアまたは8vCPU
- メモリ: 32 GB
- ディスク容量: ホストには 165 GB が推奨され、パーティション要件は次のとおりです。
  - `/opt`: 120 GiBの空き容量が必要です

    エージェントは `/opt`インストールするには `/opt/application/netapp`ディレクトリとその内容。
  - `/var`: 40 GiBの空き容量が必要です

    コンソールエージェントにはこのスペースが必要です `/var`Podman または Docker は、このディレクトリ内にコンテナを作成するように設計されているためです。具体的には、`/var/lib/containers/storage`ディレクトリと `/var/lib/docker`Docker用。このスペースでは外部マウントまたはシンボリックリンクは機能しません。

ハイパーバイザー

サポートされているオペレーティング システムを実行することが認定されているベア メタルまたはホスト型ハイパーバイザーが必要です。

オペレーティングシステムとコンテナの要件

コンソールを標準モードまたは制限モードで使用する場合、コンソール エージェントは次のオペレーティング システムでサポートされます。エージェントをインストールする前に、コンテナ オーケストレーション ツールが必要です。

オペレーティング システム	サポートされるOS バージョン	サポートされているエージェントのバージョン	必要なコンテナツール	SELinux
Red Hat Enterprise Linux		9.6 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	4.0.0 以降、コンソールが標準モードまたは制限モード	Podman バージョン 5.4.0 と podman-compose 1.5.0。 Podman の構成要件を表示する 。

オペレーティングシステム	サポートされるOSバージョン	サポートされているエージェントのバージョン	必要なコンテナツール	SELinux
強制モードまたは許可モードでサポートされます		9.1～9.4 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.9.4 と podman-compose 1.5.0。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます		8.6～8.10 • 英語版のみ。 • ホストは Red Hat Subscription Management に登録されている必要があります。登録されていない場合、ホストはエージェントのインストール中に必要なサードパーティ製ソフトウェアを更新するためにリポジトリにアクセスできません。	3.9.50 以降、コンソールが標準モードまたは制限モード	Podman バージョン 4.6.1 または 4.9.4 と podman-compose 1.0.6。 Podman の構成要件を表示する 。
強制モードまたは許可モードでサポートされます	Ubuntu		24.04 LTS	3.9.45 以降、NetApp Consoleが標準モードまたは制限モード
Docker エンジン 23.06 から 28.0.0。	サポート対象外		22.04 LTS	3.9.50以降

コンソール エージェントがリソースを管理できるようにネットワーク アクセスを設定します。ターゲット ネットワークへの接続と特定のエンドポイントへのアウトバウンド インターネット アクセスが必要です。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

Webベースの**NetApp Console**を使用する際にコンピュータから接続されるエンドポイント

Web ブラウザからコンソールにアクセスするコンピュータは、複数のエンドポイントに接続できる必要があります。コンソール エージェントを設定し、コンソールを日常的に使用するには、コンソールを使用する必要があります。

"NetAppコンソールのネットワークを準備する" 。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

オンプレミスにインストールされたコンソール エージェントは、Google Cloud 内のリソースを管理できません。 Google Cloud リソースを管理するには、Google Cloud にエージェントをインストールする必要があります。

AWS

コンソール エージェントをオンプレミスでインストールする場合、AWS に導入されたNetAppシステム (Cloud Volumes ONTAPなど) を管理するために、次の AWS エンドポイントへのネットワーク アクセスが必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
AWS サービス (amazonaws.com): • クラウドフォーメーション • エラスティックコンピューティングクラウド (EC2) • アイデンティティとアクセス管理（IAM） • キー管理サービス (KMS) • セキュリティトークンサービス（STS） • シンプルストレージサービス（S3）	AWS リソースを管理します。エンドポイントは AWS リージョンによって異なります。 "詳細についてはAWSドキュメントを参照してください"
NetApp ONTAP用の Amazon FsX: • api.workloads.netapp.com	Web ベースのコンソールは、このエンドポイントに接続して Workload Factory API と対話し、FSx for ONTAPベースのワークロードを管理および操作します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。

エンドポイント	目的
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

Azure

コンソール エージェントがオンプレミスにインストールされている場合、Azure にデプロイされたNetAppシステム (Cloud Volumes ONTAPなど) を管理するために、次の Azure エンドポイントへのネットワーク アクセスが必要です。

エンドポイント	目的
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	Azure パブリック リージョン内のリソースを管理します。
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	Azure China リージョンのリソースを管理します。

エンドポイント	目的
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

AWS または **Azure** のコンソール エージェント クラウド権限を作成する

オンプレミスのコンソールエージェントを使用して AWS または Azure のNetAppデータ サービスを使用する場合は、クラウド プロバイダーで権限を設定し、インストール後にコンソールエージェントに資格情報を追加する必要があります。

Google Cloud に存在するリソースを管理するには、Google Cloud に Console エージェントをインストールする必要があります。

AWS

コンソール エージェントがオンプレミスにインストールされている場合は、必要な権限を持つ IAM ユーザーのアクセス キーを追加して、コンソールに AWS 権限を付与する必要があります。

コンソール エージェントがオンプレミスにインストールされている場合は、この認証方法を使用する必要があります。 IAM ロールは使用できません。

手順

1. AWS コンソールにログインし、IAM サービスに移動します。
2. ポリシーを作成します。

   a. *ポリシー > ポリシーの作成*を選択します。

   b. *JSON*を選択し、その内容をコピーして貼り付けます。"コンソールエージェントのIAMポリシー" 。

   c. 残りの手順を完了してポリシーを作成します。

      使用する予定のNetAppデータ サービスによっては、2 番目のポリシーを作成する必要がある場合があります。

   標準リージョンの場合、権限は 2 つのポリシーに分散されます。 AWS の管理ポリシーの最大文字サイズ制限により、2 つのポリシーが必要になります。"コンソールエージェントのIAMポリシーの詳細" 。

3. IAM ユーザーにポリシーをアタッチします。
   - "AWSドキュメント: IAMロールの作成"
   - "AWSドキュメント: IAMポリシーの追加と削除"
4. コンソール エージェントをインストールした後、 NetApp Consoleに追加できるアクセス キーがユーザーにあることを確認します。

結果

これで、必要な権限を持つ IAM ユーザーのアクセス キーを取得できるはずです。コンソール エージェントをインストールした後、コンソールからこれらの資格情報をコンソール エージェントに関連付けます。

Azure

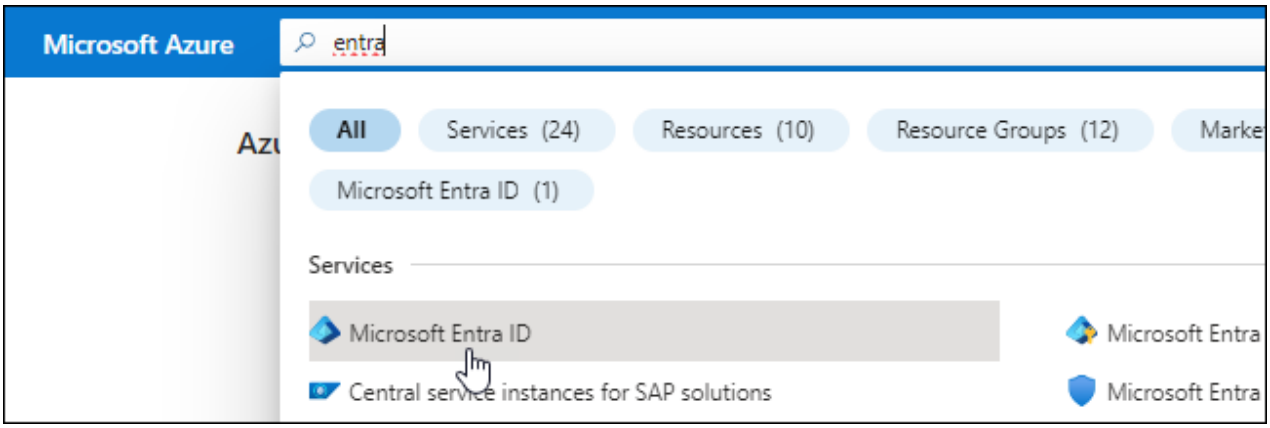
コンソール エージェントがオンプレミスでインストールされている場合は、Microsoft Entra ID でサービス プリンシパルを設定し、コンソール エージェントに必要な Azure 資格情報を取得して、コンソール エージェントに Azure 権限を付与する必要があります。

ロールベースのアクセス制御用の **Microsoft Entra** アプリケーションを作成する

1. Azure で Active Directory アプリケーションを作成し、そのアプリケーションをロールに割り当てるためのアクセス許可があることを確認します。

   詳細については、 "Microsoft Azure ドキュメント: 必要な権限"

2. Azure ポータルから、**Microsoft Entra ID** サービスを開きます。



3. メニューで*アプリ登録*を選択します。
4. *新規登録*を選択します。
5. アプリケーションの詳細を指定します。
   - 名前: アプリケーションの名前を入力します。
   - アカウント タイプ: アカウント タイプを選択します (いずれのタイプもNetApp Consoleで使用できます)。
   - リダイレクト **URI**: このフィールドは空白のままにすることができます。
6. *登録*を選択します。

   AD アプリケーションとサービス プリンシパルを作成しました。

アプリケーションをロールに割り当てる

1. カスタム ロールを作成します。

   Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

   a. の内容をコピーします"コンソールエージェントのカスタムロール権限"JSON ファイルに保存します。
   b. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

      ユーザーがCloud Volumes ONTAPシステムを作成する各 Azure サブスクリプションの ID を追加する必要があります。

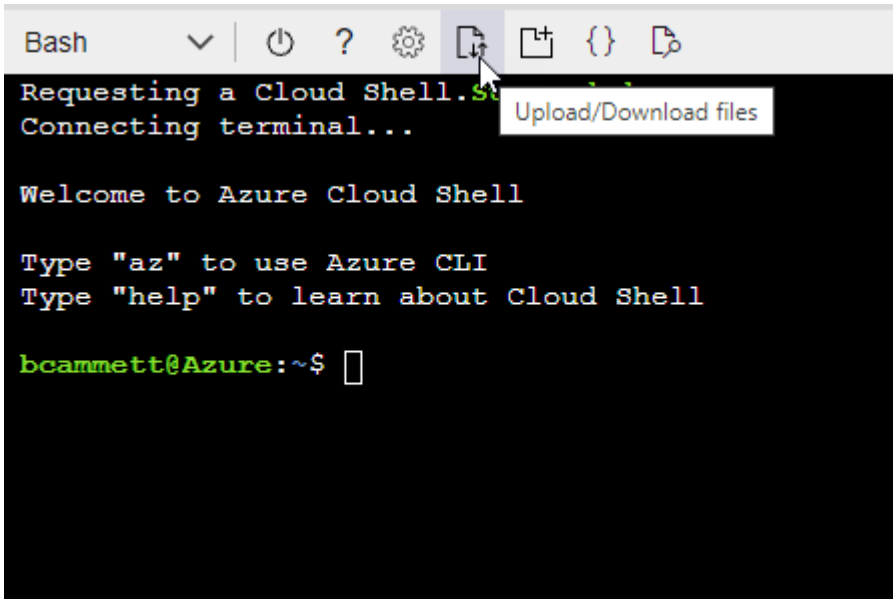
      例

```
"AssignableScopes": [
"/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
"/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
"/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
]
```

c. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

- 始める "Azure クラウド シェル"Bash 環境を選択します。
- JSON ファイルをアップロードします。



- Azure CLI を使用してカスタム ロールを作成します。

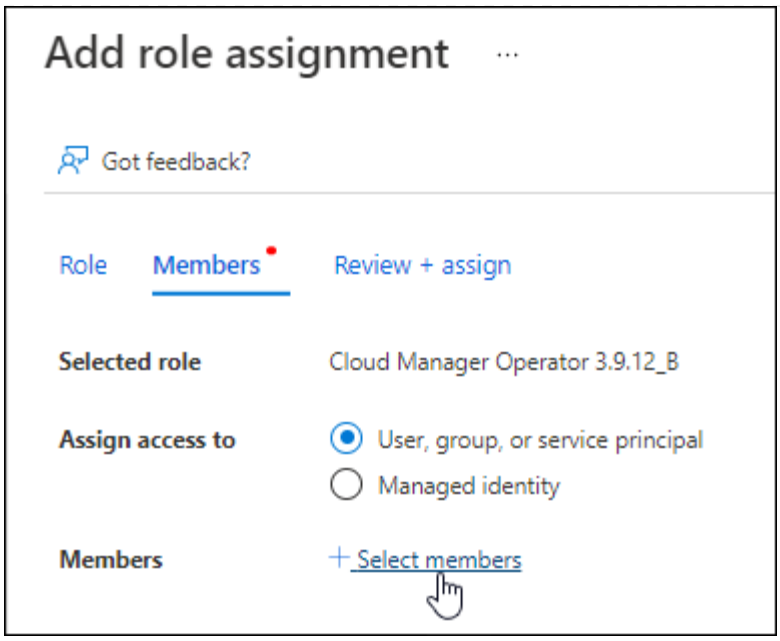
```
az role definition create --role-definition agent_Policy.json
```

これで、コンソール エージェント仮想マシンに割り当てることができる、コンソール オペレーターと呼ばれるカスタム ロールが作成されます。

2. アプリケーションをロールに割り当てます。

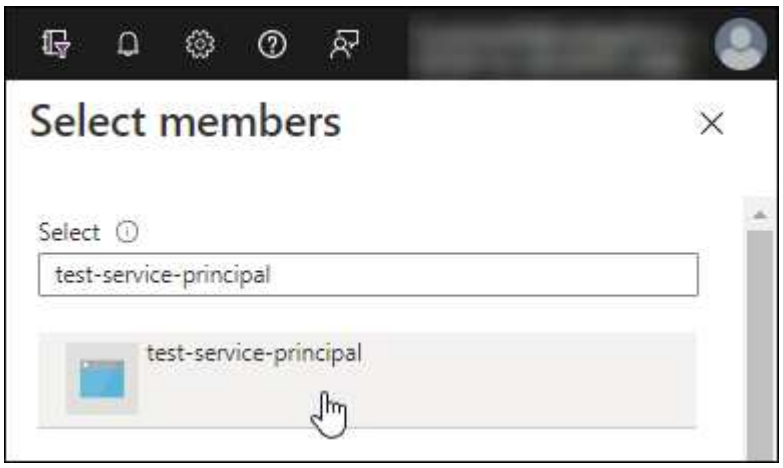
a. Azure ポータルから、サブスクリプション サービスを開きます。
b. サブスクリプションを選択します。
c. アクセス制御 **(IAM) >** 追加 **>** ロール割り当ての追加 を選択します。
d. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。
e. *メンバー*タブで、次の手順を実行します。

- *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
- *メンバーを選択*を選択します。



- アプリケーションの名前を検索します。

  次に例を示します。



- アプリケーションを選択し、[選択] を選択します。
- *次へ*を選択します。

f. *レビュー + 割り当て*を選択します。

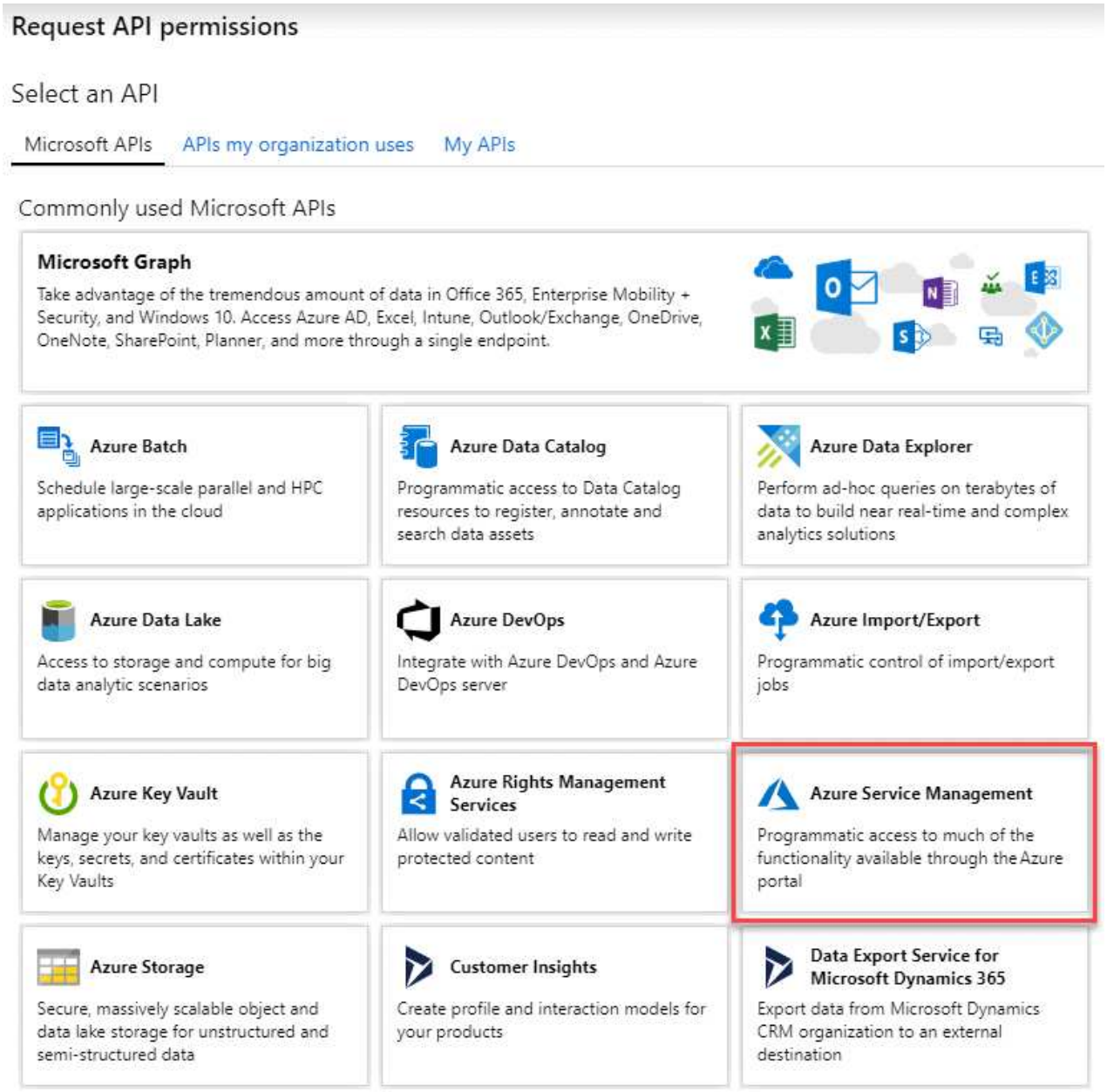
これで、サービス プリンシパルに、コンソール エージェントをデプロイするために必要な Azure アクセス許可が付与されました。

複数の Azure サブスクリプションからCloud Volumes ONTAPをデプロイする場合は、サービス プリンシパルを各サブスクリプションにバインドする必要があります。 NetApp Consoleでは、 Cloud Volumes ONTAP をデプロイするときに使用するサブスクリプションを選択できます。

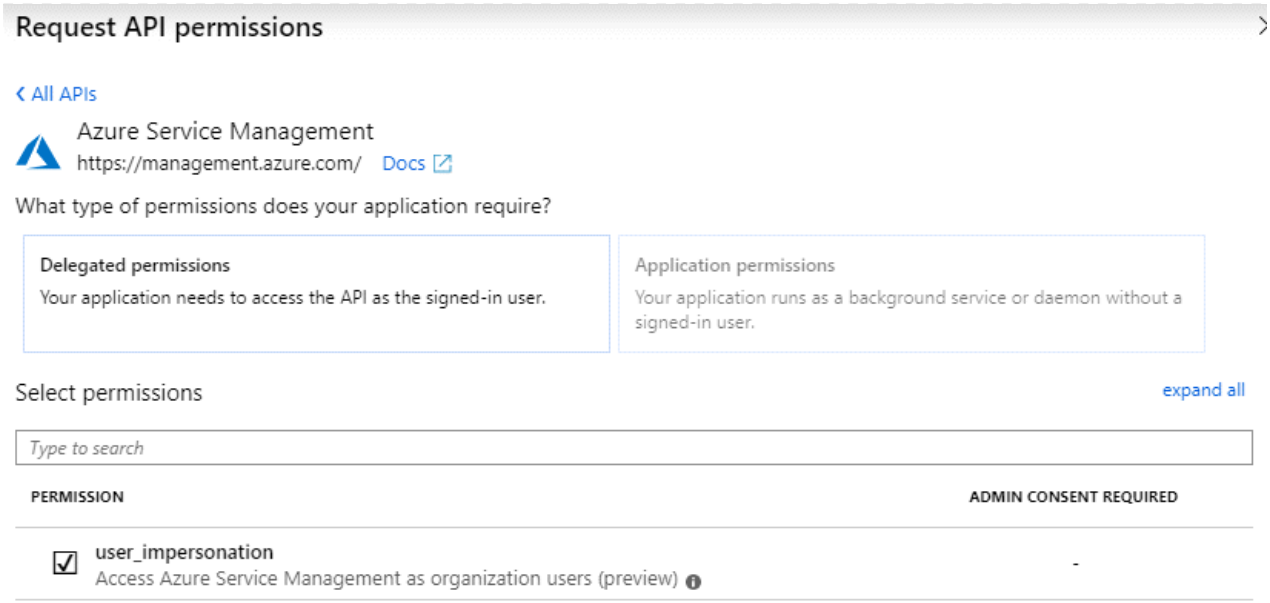
Windows Azure サービス管理 **API** 権限を追加する

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. **API** 権限 **>** 権限の追加 を選択します。

3. **Microsoft API** の下で、**Azure Service Management** を選択します。

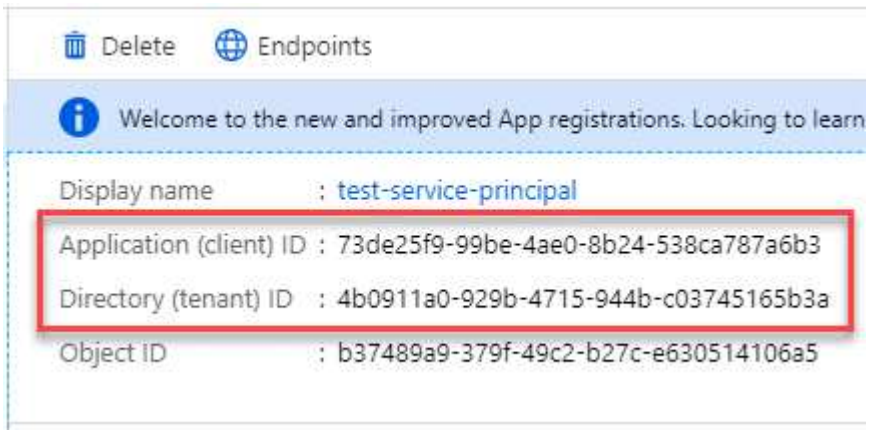


4. 組織ユーザーとして **Azure** サービス管理にアクセスする を選択し、権限の追加 を選択します。



アプリケーションのアプリケーション**ID**とディレクトリ**ID**を取得します

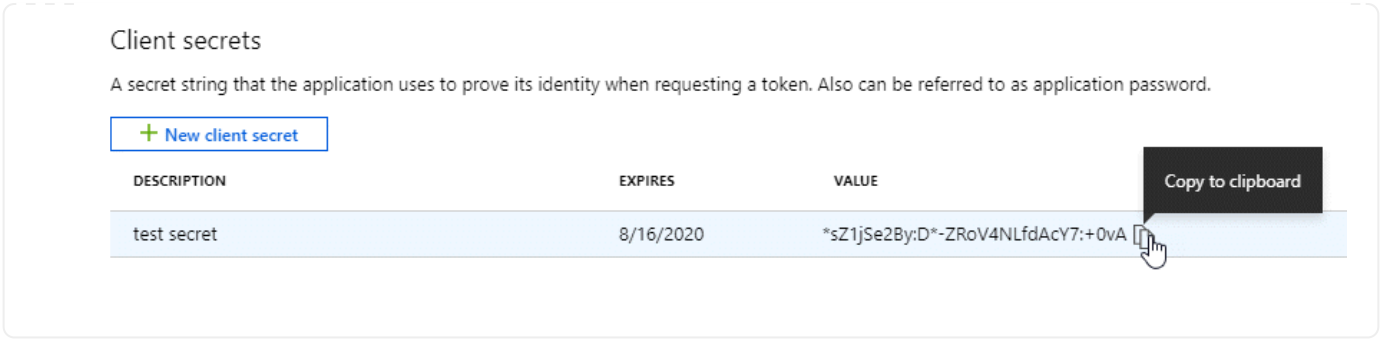
1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. アプリケーション **(**クライアント**) ID** と ディレクトリ **(**テナント**) ID** をコピーします。



Azure アカウントをコンソールに追加するときは、アプリケーションのアプリケーション (クライアント) ID とディレクトリ (テナント) ID を指定する必要があります。コンソールは ID を使用してプログラムでサインインします。

クライアントシークレットを作成する

1. **Microsoft Entra ID** サービスを開きます。
2. *アプリ登録*を選択し、アプリケーションを選択します。
3. *証明書とシークレット > 新しいクライアント シークレット*を選択します。
4. シークレットの説明と期間を指定します。
5. *追加*を選択します。
6. クライアント シークレットの値をコピーします。



コンソールエージェントを手動でインストールする

コンソール エージェントを手動でインストールする場合は、要件を満たすようにマシン環境を準備する必要があります。 Linux マシンが必要であり、Linux オペレーティング システムに応じて Podman または Docker をインストールする必要があります。

Podmanまたは**Docker Engine**をインストールする

オペレーティング システムに応じて、エージェントをインストールする前に Podman または Docker Engine のいずれかが必要になります。

- Red Hat Enterprise Linux 8 および 9 には Podman が必要です。

  サポートされているPodmanのバージョンを表示する 。

- Ubuntu には Docker Engine が必要です。

  サポートされている Docker エンジンのバージョンを表示する 。

例 **4.** 手順

ポッドマン

Podman をインストールして設定するには、次の手順に従います。

- podman.socket サービスを有効にして起動します
- Python3をインストールする
- podman-compose パッケージ バージョン 1.0.6 をインストールします。
- podman-composeをPATH環境変数に追加する
- Red Hat Enterprise Linux を使用している場合は、Podman バージョンが CNI ではなく Netavark Aardvark DNS を使用していることを確認してください。

DNS ポートの競合を避けるために、エージェントをインストールした後、aardvark-dns ポート (デフォルト: 53) を調整します。指示に従ってポートを構成します。

手順

1. ホストに podman-docker パッケージがインストールされている場合は削除します。

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Podman をインストールします。

   Podman は、公式の Red Hat Enterprise Linux リポジトリから入手できます。

   a. Red Hat Enterprise Linux 9.6 の場合:

   ```
   sudo dnf install podman-5:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   b. Red Hat Enterprise Linux 9.1 から 9.4 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

   <version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

   c. Red Hat Enterprise Linux 8 の場合:

   ```
   sudo dnf install podman-4:<version>
   ```

<version> は、インストールする Podman のサポートされているバージョンです。サポートされているPodmanのバージョンを表示する 。

3. podman.socket サービスを有効にして起動します。

```
sudo systemctl enable --now podman.socket
```

4. python3 をインストールします。

```
sudo dnf install python3
```

5. システムにまだインストールされていない場合は、EPEL リポジトリ パッケージをインストールします。

   podman-compose は、Extra Packages for Enterprise Linux (EPEL) リポジトリから入手できるため、この手順は必須です。

6. Red Hat Enterprise 9 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

   +

   a. podman-compose パッケージ 1.5.0 をインストールします。

```
sudo dnf install podman-compose-1.5.0
```

7. Red Hat Enterprise Linux 8 を使用している場合:

   a. EPEL リポジトリ パッケージをインストールします。

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

   b. podman-compose パッケージ 1.0.6 をインストールします。

```
sudo dnf install podman-compose-1.0.6
```

使用して `dnf install`コマンドは、PATH 環境変数に podman-compose を追加するための要件を満たしています。インストールコマンドは、すでにインストールされているpodman-composeを/usr/binに追加します。 `secure_path`ホスト上のオプション。

c. Red Hat Enterprise Linux 8 を使用している場合は、Podman バージョンが CNI ではなく Aardvark DNS を備えた NetAvark を使用していることを確認します。

i. 次のコマンドを実行して、networkBackend が CNI に設定されているかどうかを確認します。

```
podman info | grep networkBackend
```

ii. networkBackendが `CNI`、それを変更する必要があります `netavark`。

iii. インストール `netavark`そして `aardvark-dns`次のコマンドを使用します。

```
dnf install aardvark-dns netavark
```

iv. 開く `/etc/containers/containers.conf`ファイルを編集し、network_backend オプションを変更して、「cni」の代わりに「netavark」を使用します。

もし `/etc/containers/containers.conf`存在しない場合は、設定を変更してください `/usr/share/containers/containers.conf`。

v. podman を再起動します。

```
systemctl restart podman
```

vi. 次のコマンドを使用して、networkBackend が「netavark」に変更されていることを確認します。

```
podman info | grep networkBackend
```

Dockerエンジン

Docker のドキュメントに従って Docker Engine をインストールします。

手順

1. "Dockerからのインストール手順を見る"

   サポートされている Docker エンジン バージョンをインストールするには、手順に従ってください。最新バージョンはコンソールでサポートされていないため、インストールしないでください。

2. Docker が有効になっていて実行されていることを確認します。

```
sudo systemctl enable docker && sudo systemctl start docker
```

コンソールエージェントを手動でインストールする

オンプレミスの既存の Linux ホストにコンソール エージェント ソフトウェアをダウンロードしてインストールします。

開始する前に

次のものが必要です:

- コンソール エージェントをインストールするためのルート権限。
- コンソール エージェントからのインターネット アクセスにプロキシが必要な場合のプロキシ サーバーの詳細。

  インストール後にプロキシ サーバーを構成するオプションがありますが、これを行うにはコンソール エージェントを再起動する必要があります。

- プロキシ サーバーが HTTPS を使用する場合、またはプロキシがインターセプト プロキシである場合は、CA 署名証明書。

> コンソール エージェントを手動でインストールする場合、透過プロキシ サーバーの証明書を設定することはできません。透過プロキシ サーバーの証明書を設定する必要がある場合は、インストール後にメンテナンス コンソールを使用する必要があります。詳細はこちら "エージェントメンテナンスコンソール"。

タスク概要

インストール後、新しいバージョンが利用可能な場合、コンソール エージェントは自動的に更新されます。

手順

1. ホストに *http_proxy* または *https_proxy* システム変数が設定されている場合は、それらを削除します。

   ```
   unset http_proxy
   unset https_proxy
   ```

   これらのシステム変数を削除しないと、インストールは失敗します。

2. コンソール エージェント ソフトウェアをダウンロードし、Linux ホストにコピーします。 NetApp ConsoleまたはNetAppサポート サイトからダウンロードできます。

   - NetApp Console: エージェント > 管理 > エージェントのデプロイ > オンプレミス > 手動インストールに移動します。

     エージェント インストーラー ファイルのダウンロードまたはファイルへの URL を選択します。

   - NetAppサポート サイト (コンソールにまだアクセスできない場合に必要) "NetAppサポート サイト" 、

3. スクリプトを実行するための権限を割り当てます。

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

<version> は、ダウンロードしたコンソール エージェントのバージョンです。

4. Government Cloud 環境にインストールする場合は、構成チェックを無効にします。"手動インストールの構成チェックを無効にする方法を説明します。"
5. インストール スクリプトを実行します。

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy server> --cacert <path and file name of a CA-signed certificate>
```

ネットワークでインターネットアクセスにプロキシが必要な場合は、プロキシ情報を追加する必要があります。インストール中に明示的にプロキシを追加できます。 `--proxy`および `--cacert`パラメータはオプションであり、追加を要求されることはありません。明示的なプロキシ サーバがある場合は、示されているようにパラメータを入力する必要があります。

透過プロキシを設定する場合は、インストール後に設定できます。 "エージェントメンテナンスコンソールについて学ぶ"

+

CA 署名証明書を使用して明示的なプロキシ サーバーを構成する例を次に示します。

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+
`--proxy` 次のいずれかの形式を使用して、Console エージェントが HTTP または HTTPS プロキシ サーバを使用するように設定します：

+ * http://address:port * http://user-name:password@address:port * http://domain-name%92user-name:password@address:port * https://address:port * https://user-name:password@address:port * https://domain-name%92user-name:password@address:port

+ 以下の点に注意してください：

+ ユーザーは、ローカル ユーザーまたはドメイン ユーザーにすることができます。 ドメイン ユーザーの場合は、上記のように \ の ASCII コードを使用する必要があります。 **Console** エージェントは、**@** 文字を含むユーザー名またはパスワードをサポートしていません。 パスワードに次の特殊文字が含まれている場合は、その特殊文字の前にバックスラッシュを付けてエスケープする必要があります：& または !

+ 例：

+ http://bxpproxyuser:netapp1\!@address:3128

1. Podman を使用した場合は、aardvark-dns ポートを調整する必要があります。

   a. コンソール エージェント仮想マシンに SSH で接続します。

   b. podman */usr/share/containers/containers.conf* ファイルを開き、Aardvark DNS サービス用に選択したポートを変更します。たとえば、54 に変更します。

      ```
      vi /usr/share/containers/containers.conf
      ```

      例えば：

      ```
      # Port to use for dns forwarding daemon with netavark in rootful bridge
      # mode and dns enabled.
      # Using an alternate port might be useful if other DNS services should
      # run on the machine.
      #
      dns_bind_port = 54
      ```

   a. コンソール エージェント仮想マシンを再起動します。

次は何？

NetApp Console内でコンソール エージェントを登録する必要があります。

NetApp Consoleにコンソールエージェントを登録する

コンソールにログインし、コンソール エージェントを組織に関連付けます。ログイン方法は、コンソールを使用しているモードによって異なります。コンソールを標準モードで使用している場合は、SaaS Web サイトからログインします。コンソールを制限モードで使用している場合は、コンソール エージェント ホストからローカルにログインします。

手順

1. Web ブラウザを開き、コンソール エージェント ホストの URL を入力します。

   コンソール ホスト URL は、ホストの構成に応じて、ローカルホスト、プライベート IP アドレス、またはパブリック IP アドレスになります。たとえば、コンソール エージェントがパブリック IP アドレスのないパブリック クラウドにある場合は、コンソール エージェント ホストに接続しているホストのプライベート IP アドレスを入力する必要があります。

2. サインアップまたはログインしてください。

3. ログイン後、コンソールを設定します。

   a. コンソール エージェントに関連付けるコンソール組織を指定します。

   b. システムの名前を入力します。

   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      コンソール エージェントがオンプレミスにインストールされている場合、制限モードはサポートされません。

d. *始めましょう*を選択します。

NetApp Consoleにクラウドプロバイダーの資格情報を提供する

コンソール エージェントをインストールしてセットアップしたら、コンソール エージェントが AWS または Azure でアクションを実行するために必要な権限を持つように、クラウド資格情報を追加します。

AWS

開始する前に

これらの AWS 認証情報を作成したばかりの場合は、使用可能になるまでに数分かかることがあります。資格情報をコンソールに追加する前に、数分待ってください。

手順

1. *管理 > 資格情報*を選択します。
2. *組織の資格情報*を選択します。
3. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: *Amazon Web Services > エージェントを選択します。
   b. 資格情報の定義: AWS アクセスキーとシークレットキーを入力します。
   c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。
   d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

これで、 "NetApp Console"コンソール エージェントの使用を開始します。

Azure

開始する前に

これらの Azure 資格情報を作成したばかりの場合は、使用可能になるまでに数分かかることがあります。コンソール エージェントに資格情報を追加する前に、数分間お待ちください。

手順

1. *管理 > 資格情報*を選択します。
2. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: **Microsoft Azure >** エージェント を選択します。
   b. 資格情報の定義: 必要な権限を付与する Microsoft Entra サービス プリンシパルに関する情報を入力します。
      - アプリケーション（クライアント）ID
      - ディレクトリ（テナント）ID
      - クライアントシークレット
   c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。
   d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

結果

これで、コンソール エージェントには、ユーザーに代わって Azure でアクションを実行するために必要なアクセス許可が付与されました。これで、 "NetApp Console"コンソール エージェントの使用を開始します。

VCenter を使用してオンプレミスにコンソール エージェントをインストールする

VMWare ユーザーの場合は、OVA を使用して VCenter にコンソール エージェントをインストールできます。 OVA のダウンロードまたは URL は、NetApp Consoleから入手できます。

VCenter ツールと共にコンソール エージェントをインストールすると、VM Web コンソールを使用してメンテナンス タスクを実行できます。"エージェントの VM コンソールの詳細について説明します。"

コンソールエージェントのインストールの準備

インストールする前に、VM ホストが要件を満たしており、コンソール エージェントがインターネットおよび対象のネットワークにアクセスできることを確認してください。 NetAppデータ サービスまたはCloud Volumes ONTAP を使用するには、コンソール エージェントがユーザーに代わってアクションを実行できるように、クラウド プロバイダの資格情報を作成します。

コンソールエージェントホストの要件を確認する

コンソール エージェントをインストールする前に、ホスト マシンがインストール要件を満たしていることを確認してください。

- CPU: 8コアまたは8vCPU
- メモリ: 32 GB
- ディスク容量: 165 GB (シックプロビジョニング)
- vSphere 7.0以降
- ESXi ホスト 7.03 以上

エージェントを ESXi ホストに直接インストールするのではなく、vCenter 環境にインストールします。

コンソールエージェントのネットワークアクセスを設定する

ネットワーク管理者と協力して、コンソール エージェントが必要なエンドポイントへの送信アクセスと対象ネットワークへの接続を持っていることを確認します。

ターゲットネットワークへの接続

コンソール エージェントには、システムを作成および管理する予定の場所へのネットワーク接続が必要です。たとえば、オンプレミス環境にCloud Volumes ONTAPシステムまたはストレージ システムを作成する予定のネットワークなどです。

アウトバウンドインターネットアクセス

コンソール エージェントを展開するネットワークの場所には、特定のエンドポイントに接続するための送信インターネット接続が必要です。

Webベースの**NetApp Console**を使用する際にコンピュータから接続されるエンドポイント

Web ブラウザからコンソールにアクセスするコンピュータは、複数のエンドポイントに接続できる必要があります。コンソール エージェントを設定し、コンソールを日常的に使用するには、コンソールを使用す

る必要があります。

"NetAppコンソールのネットワークを準備する" 。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

オンプレミスにコンソール エージェントがインストールされている場合、Google Cloud のリソースを管理することはできません。 Google Cloud リソースを管理するには、Google Cloud にエージェントをインストールします。

AWS

コンソール エージェントをオンプレミスでインストールする場合、AWS に導入されたNetAppシステム (Cloud Volumes ONTAPなど) を管理するために、次の AWS エンドポイントへのネットワーク アクセスが必要です。

コンソールエージェントから接続されたエンドポイント

コンソール エージェントは、日常業務でパブリック クラウド環境内のリソースとプロセスを管理するために、次のエンドポイントに接続するために、送信インターネット アクセスを必要とします。

以下にリストされているエンドポイントはすべて CNAME エントリです。

エンドポイント	目的
AWS サービス (amazonaws.com): • クラウドフォーメーション • エラスティックコンピューティングクラウド (EC2) • アイデンティティとアクセス管理（IAM） • キー管理サービス (KMS) • セキュリティトークンサービス（STS） • シンプルストレージサービス（S3）	AWS リソースを管理します。エンドポイントは AWS リージョンによって異なります。 "詳細についてはAWSドキュメントを参照してください"
NetApp ONTAP用の Amazon FsX: • api.workloads.netapp.com	Web ベースのコンソールは、このエンドポイントに接続して Workload Factory API と対話し、FSx for ONTAPベースのワークロードを管理および操作します。
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。

エンドポイント	目的
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

Azure

コンソール エージェントがオンプレミスにインストールされている場合、Azure にデプロイされたNetAppシステム (Cloud Volumes ONTAPなど) を管理するために、次の Azure エンドポイントへのネットワーク アクセスが必要です。

エンドポイント	目的
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	Azure パブリック リージョン内のリソースを管理します。
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	Azure China リージョンのリソースを管理します。

エンドポイント	目的
https://mysupport.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信します。
https://signin.b2c.netapp.com	NetAppサポート サイト (NSS) の資格情報を更新したり、 NetApp Consoleに新しい NSS 資格情報を追加したりします。
https://support.netapp.com	ライセンス情報を取得し、AutoSupportメッセージをNetAppサポートに送信し、 Cloud Volumes ONTAPのソフトウェアアップデートを受信します。
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	NetApp Console内で機能とサービスを提供します。
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	コンソール エージェントのアップグレード用のイメージを取得します。 • 新しいエージェントを展開すると、検証チェックによって現在のエンドポイントへの接続がテストされます。使用する場合"以前のエンドポイント"、検証チェックは失敗します。この失敗を回避するには、検証チェックをスキップします。 以前のエンドポイントも引き続きサポートされますが、NetApp、ファイアウォールルールをできるだけ早く現在のエンドポイントに更新することをお勧めします。"エンドポイントリストを更新する方法を学ぶ" 。 • ファイアウォールの現在のエンドポイントに更新すると、既存のエージェントは引き続き動作します。

プロキシ サーバ

NetApp は明示的プロキシ構成と透過的プロキシ構成の両方をサポートしています。透過プロキシを使用している場合は、プロキシ サーバーの証明書のみを提供する必要があります。明示的なプロキシを使用している場合は、IP アドレスと資格情報も必要になります。

- IPアドレス
- Credentials
- HTTPS証明書

ポート

ユーザーが開始した場合、またはCloud Volumes ONTAPからNetAppサポートにAutoSupportメッセージを送信するためのプロキシとして使用された場合を除いて、コンソール エージェントへの着信トラフィックはありません。

- HTTP (80) と HTTPS (443) は、まれにしか使用されないローカル UI へのアクセスを提供します。
- SSH（22）は、トラブルシューティングのためにホストに接続する必要がある場合にのみ必要です。
- アウトバウンド インターネット接続が利用できないサブネットにCloud Volumes ONTAPシステムを展開する場合は、ポート 3128 経由のインバウンド接続が必要です。

  Cloud Volumes ONTAPシステムにAutoSupportメッセージを送信するためのアウトバウンド インターネット接続がない場合、コンソールは、コンソール エージェントに含まれているプロキシ サーバーを使用するようにそれらのシステムを自動的に構成します。唯一の要件は、コンソール エージェントのセキュリティ グループがポート 3128 経由の受信接続を許可していることを確認することです。コンソール エージェントを展開した後、このポートを開く必要があります。

NTP を有効にする

NetApp Data Classificationを使用して企業のデータ ソースをスキャンする予定の場合は、システム間で時刻が同期されるように、コンソール エージェントとNetApp Data Classificationシステムの両方で Network Time Protocol (NTP) サービスを有効にする必要があります。 "NetAppデータ分類の詳細"

AWS または **Azure** のコンソール エージェント クラウド権限を作成する

オンプレミスのコンソールエージェントを使用して AWS または Azure のNetAppデータ サービスを使用する場合は、インストール後にコンソールエージェントに資格情報を追加できるように、クラウド プロバイダーで権限を設定する必要があります。

オンプレミスにコンソール エージェントがインストールされている場合、Google Cloud のリソースを管理することはできません。 Google Cloud リソースを管理するには、Google Cloud にエージェントをインストールする必要があります。

AWS

オンプレミスのコンソールエージェントの場合は、IAM ユーザーアクセスキーを追加して AWS 権限を付与します。

オンプレミスのコンソール エージェントには IAM ユーザー アクセス キーを使用します。オンプレミスのコンソール エージェントでは IAM ロールはサポートされていません。

手順

1. AWS コンソールにログインし、IAM サービスに移動します。
2. ポリシーを作成します。

   a. *ポリシー > ポリシーの作成*を選択します。

   b. *JSON*を選択し、その内容をコピーして貼り付けます。"コンソールエージェントのIAMポリシー" 。

   c. 残りの手順を完了してポリシーを作成します。

      使用する予定のNetAppデータ サービスによっては、2 番目のポリシーを作成する必要がある場合があります。

   標準リージョンの場合、権限は 2 つのポリシーに分散されます。 AWS の管理ポリシーの最大文字サイズ制限により、2 つのポリシーが必要になります。"コンソールエージェントのIAMポリシーの詳細" 。

3. IAM ユーザーにポリシーをアタッチします。
   - "AWSドキュメント: IAMロールの作成"
   - "AWSドキュメント: IAMポリシーの追加と削除"
4. コンソール エージェントをインストールした後、 NetApp Consoleに追加できるアクセス キーがユーザーにあることを確認します。

結果

これで、必要な権限を持つ IAM ユーザー アクセス キーを取得できるはずです。コンソール エージェントをインストールした後、コンソールからこれらの認証情報をコンソール エージェントに関連付けます。

Azure

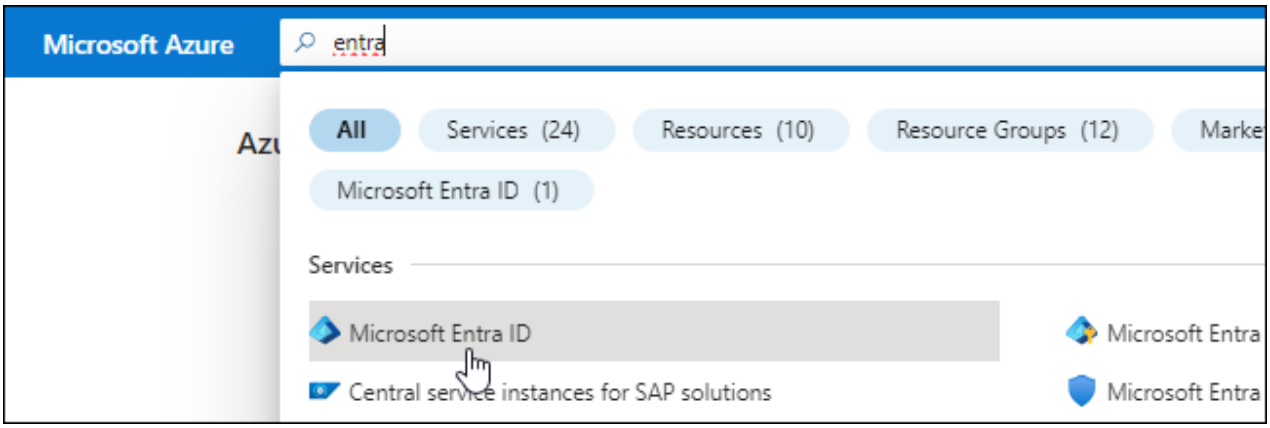
コンソール エージェントがオンプレミスでインストールされている場合は、Microsoft Entra ID でサービス プリンシパルを設定し、コンソール エージェントに必要な Azure 資格情報を取得して、コンソール エージェントに Azure 権限を付与する必要があります。

ロールベースのアクセス制御用の **Microsoft Entra** アプリケーションを作成する

1. Azure で Active Directory アプリケーションを作成し、そのアプリケーションをロールに割り当てるためのアクセス許可があることを確認します。

   詳細については、 "Microsoft Azure ドキュメント: 必要な権限"

2. Azure ポータルから、**Microsoft Entra ID** サービスを開きます。



3. メニューで*アプリ登録*を選択します。
4. *新規登録*を選択します。
5. アプリケーションの詳細を指定します。
   - 名前: アプリケーションの名前を入力します。
   - アカウント タイプ: アカウント タイプを選択します (いずれのタイプもNetApp Consoleで使用できます)。
   - リダイレクト **URI**: このフィールドは空白のままにすることができます。
6. *登録*を選択します。

   AD アプリケーションとサービス プリンシパルを作成しました。

アプリケーションをロールに割り当てる

1. カスタム ロールを作成します。

   Azure ポータル、Azure PowerShell、Azure CLI、または REST API を使用して、Azure カスタム ロールを作成できます。次の手順は、Azure CLI を使用してロールを作成する方法を示しています。別の方法をご希望の場合は、 "Azureドキュメント"

   a. の内容をコピーします"コンソールエージェントのカスタムロール権限"JSON ファイルに保存します。
   b. 割り当て可能なスコープに Azure サブスクリプション ID を追加して、JSON ファイルを変更します。

      ユーザーがCloud Volumes ONTAPシステムを作成する各 Azure サブスクリプションの ID を追加する必要があります。

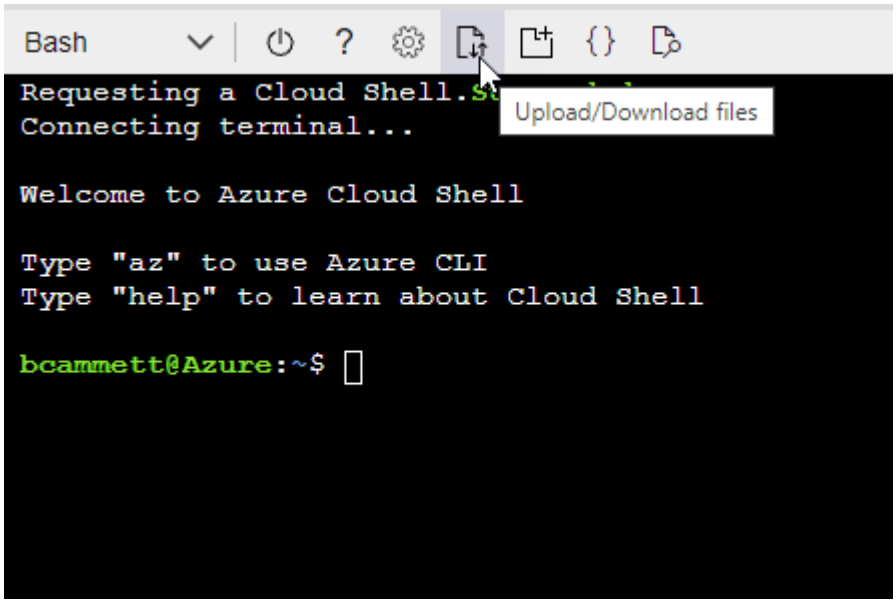
      例

```
"AssignableScopes": [
"/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
"/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
"/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
]
```

c. JSON ファイルを使用して、Azure でカスタム ロールを作成します。

次の手順では、Azure Cloud Shell で Bash を使用してロールを作成する方法について説明します。

- 始める "Azure クラウド シェル"Bash 環境を選択します。
- JSON ファイルをアップロードします。



- Azure CLI を使用してカスタム ロールを作成します。

```
az role definition create --role-definition agent_Policy.json
```

これで、コンソール エージェント仮想マシンに割り当てることができる、コンソール オペレーターと呼ばれるカスタム ロールが作成されます。

2. アプリケーションをロールに割り当てます。

a. Azure ポータルから、サブスクリプション サービスを開きます。

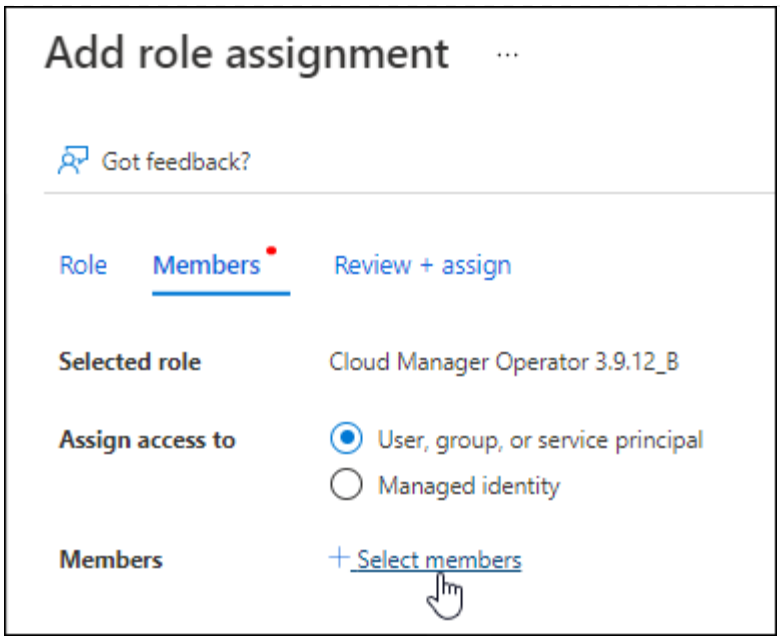
b. サブスクリプションを選択します。

c. アクセス制御 **(IAM) >** 追加 **>** ロール割り当ての追加 を選択します。

d. *役割*タブで、*コンソールオペレーター*役割を選択し、*次へ*を選択します。

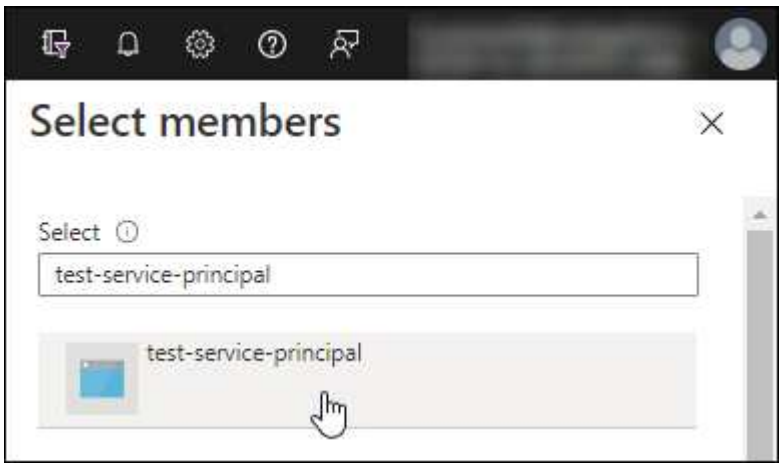
e. *メンバー*タブで、次の手順を実行します。

- *ユーザー、グループ、またはサービス プリンシパル*を選択したままにします。
- *メンバーを選択*を選択します。



- アプリケーションの名前を検索します。

  次に例を示します。



- アプリケーションを選択し、[選択] を選択します。
- *次へ*を選択します。

f. *レビュー + 割り当て*を選択します。

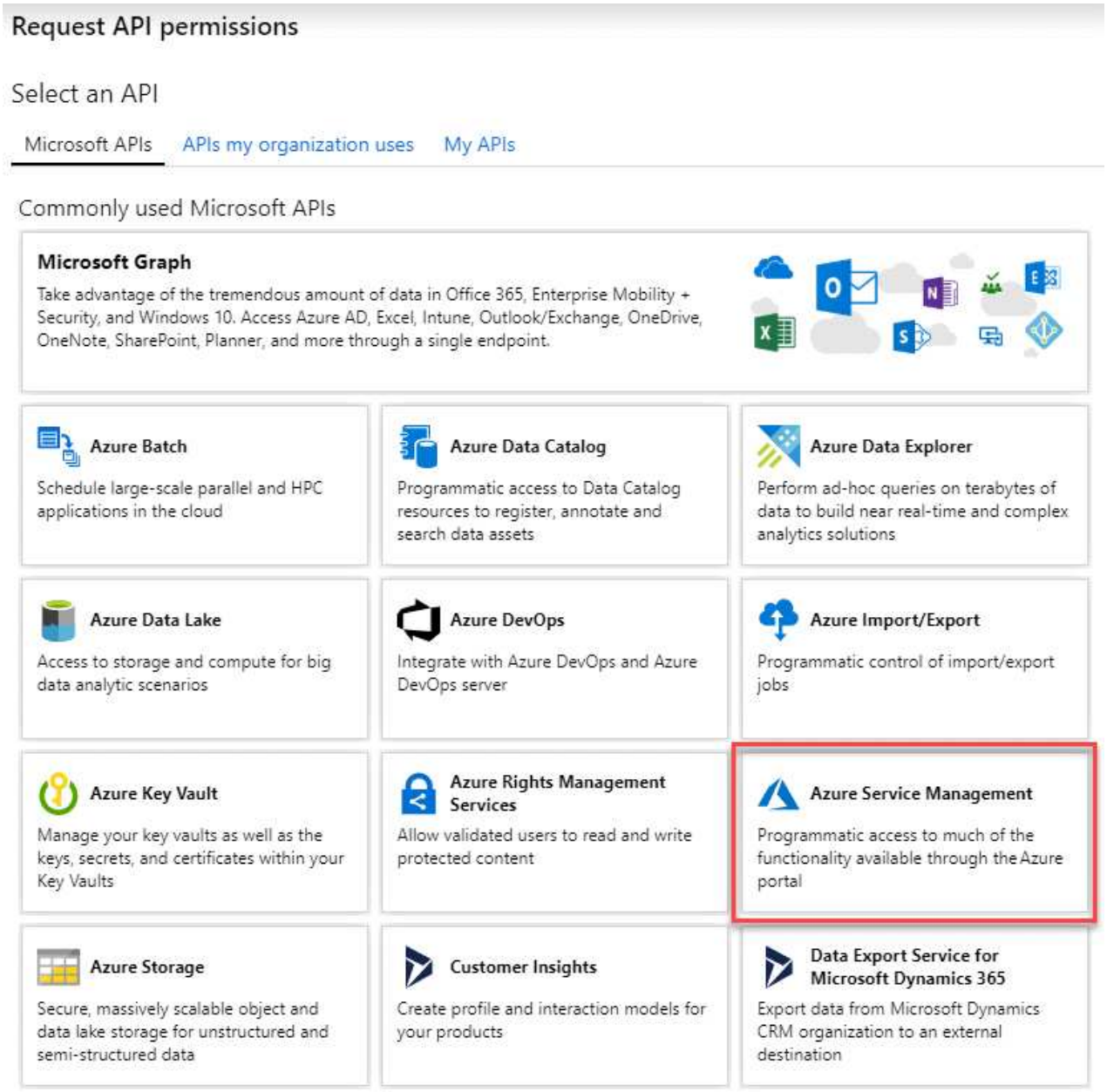
これで、サービス プリンシパルに、コンソール エージェントをデプロイするために必要な Azure アクセス許可が付与されました。

複数の Azure サブスクリプションからCloud Volumes ONTAPをデプロイする場合は、サービス プリンシパルを各サブスクリプションにバインドする必要があります。 NetApp Consoleでは、 Cloud Volumes ONTAP をデプロイするときに使用するサブスクリプションを選択できます。

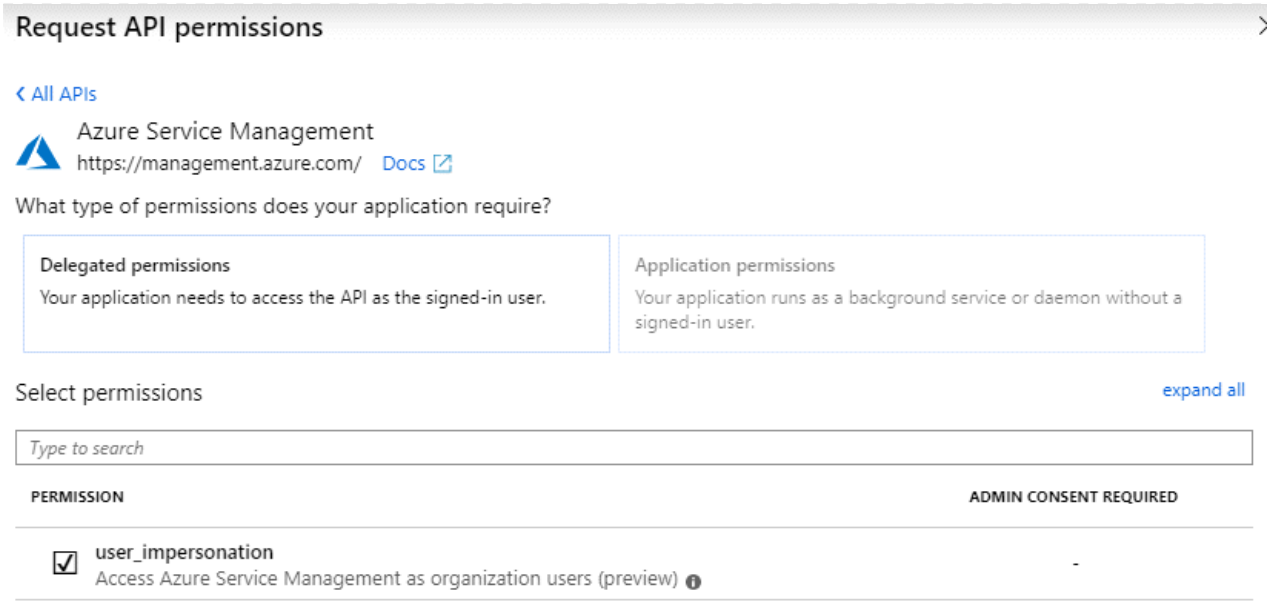
Windows Azure サービス管理 **API** 権限を追加する

1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. **API** 権限 **>** 権限の追加 を選択します。

3. **Microsoft API** の下で、**Azure Service Management** を選択します。

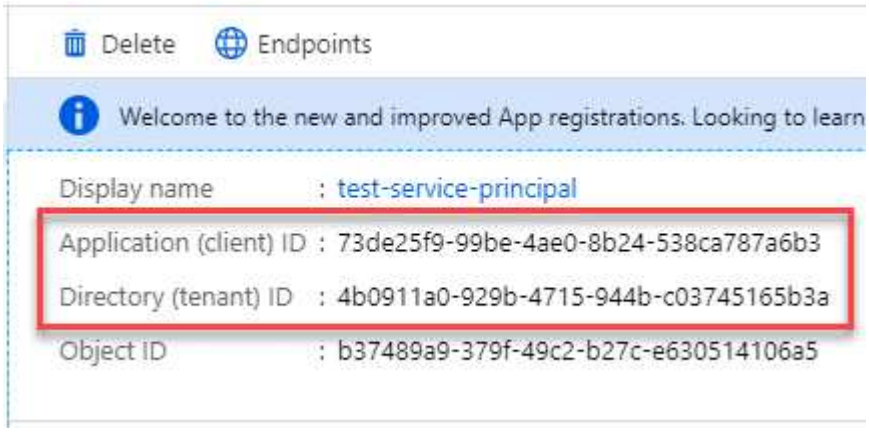


4. 組織ユーザーとして **Azure** サービス管理にアクセスする を選択し、権限の追加 を選択します。



アプリケーションのアプリケーション**ID**とディレクトリ**ID**を取得します

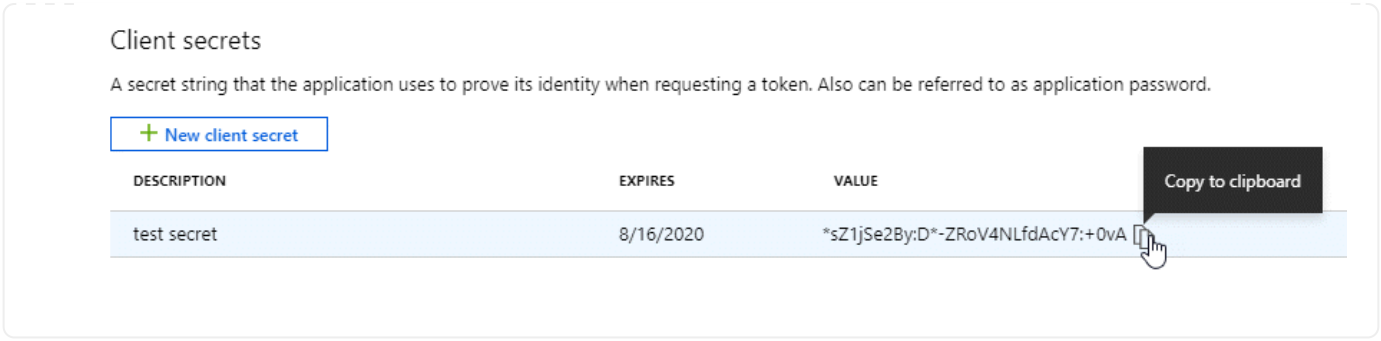
1. **Microsoft Entra ID** サービスで、アプリの登録 を選択し、アプリケーションを選択します。
2. アプリケーション **(**クライアント**) ID** と ディレクトリ **(**テナント**) ID** をコピーします。



Azure アカウントをコンソールに追加するときは、アプリケーションのアプリケーション (クライアント) ID とディレクトリ (テナント) ID を指定する必要があります。コンソールは ID を使用してプログラムでサインインします。

クライアントシークレットを作成する

1. **Microsoft Entra ID** サービスを開きます。
2. *アプリ登録*を選択し、アプリケーションを選択します。
3. *証明書とシークレット > 新しいクライアント シークレット*を選択します。
4. シークレットの説明と期間を指定します。
5. *追加*を選択します。
6. クライアント シークレットの値をコピーします。



VCenter環境にコンソールエージェントをインストールする

NetApp は、VCenter 環境へのコンソール エージェントのインストールをサポートしています。 OVA ファイルには、VMware 環境に展開できる事前構成済みの VM イメージが含まれています。ファイルのダウンロードまたは URL の展開は、 NetApp Consoleから直接行えます。コンソール エージェント ソフトウェアと自己署名証明書が含まれています。

OVAをダウンロードするかURLをコピーしてください

OVA をダウンロードするか、 NetApp Consoleから OVA URL を直接コピーします。

1. *管理 > エージェント*を選択します。
2. *概要*ページで、*エージェントのデプロイ > オンプレミス*を選択します。
3. *OVA付き*を選択してください。
4. OVA をダウンロードするか、VCenter で使用する URL をコピーするかを選択します。

VCenterにエージェントを展開する

エージェントを展開するには、VCenter 環境にログインします。

手順

1. 環境で必要な場合は、自己署名証明書を信頼できる証明書にアップロードします。インストール後にこの証明書を置き換えます。"自己署名証明書を置き換える方法を学びます。"
2. コンテンツ ライブラリまたはローカル システムから OVA を展開します。

ローカルシステムから	コンテンツライブラリから
a. 右クリックして、[OVF テンプレートのデプロイ…] を選択します。b. URL から OVA ファイルを選択するか、その場所を参照して、[次へ] を選択します。	a. コンテンツライブラリに移動し、コンソールエージェントOVAを選択します。b. アクション > *このテンプレートから新しいVM*を選択します。

3. OVF テンプレートのデプロイ ウィザードを完了して、コンソール エージェントをデプロイします。
4. VM の名前とフォルダーを選択し、[次へ] を選択します。
5. コンピューティング リソースを選択し、[次へ] を選択します。
6. テンプレートの詳細を確認し、[次へ] を選択します。
7. ライセンス契約に同意し、[次へ] を選択します。

8. 使用するプロキシ構成のタイプ（明示的プロキシ、透過プロキシ、またはプロキシなし）を選択します。
9. VM を展開するデータストアを選択し、[次へ] を選択します。ホストの要件を満たしていることを確認してください。
10. VM を接続するネットワークを選択し、[次へ] を選択します。ネットワークが IPv4 であり、必要なエンドポイントへのアウトバウンド インターネット アクセスできることを確認します。
11. *テンプレートのカスタマイズ*ウィンドウで、次のフィールドに入力します。
    - プロキシ情報
      - 明示的なプロキシを選択した場合は、プロキシ サーバーのホスト名または IP アドレスとポート番号、およびユーザー名とパスワードを入力します。
      - 透過プロキシを選択した場合は、それぞれの証明書をアップロードします。
    - 仮想マシンの構成
      - 構成チェックをスキップ: このチェックボックスはデフォルトでオフになっており、エージェントはネットワーク アクセスを検証するために構成チェックを実行します。
        - NetApp、インストールにエージェントの構成チェックが含まれるように、このボックスをオフのままにしておくことを推奨しています。構成チェックでは、エージェントが必要なエンドポイントへのネットワーク アクセス権を持っているかどうかを検証します。接続の問題によりデプロイメントが失敗した場合は、エージェント ホストから検証レポートとログにアクセスできます。場合によっては、エージェントがネットワークにアクセスできることが確実な場合は、チェックをスキップすることもできます。例えば、まだ"以前のエンドポイント"エージェントのアップグレードに使用すると、検証が失敗し、エラーが発生します。これを回避するには、検証チェックなしでインストールするためのチェックボックスをオンにします。"エンドポイントリストを更新する方法を学ぶ" 。
      - メンテナンスパスワード: `maint`エージェントメンテナンスコンソールへのアクセスを許可するユーザー。
      - **NTP** サーバー: 時刻同期用の 1 つ以上の NTP サーバーを指定します。
      - ホスト名: この VM のホスト名を設定します。検索ドメインを含めることはできません。たとえば、console10.searchdomain.company.com の FQDN は console10 と入力する必要があります。
      - プライマリ **DNS**: 名前解決に使用するプライマリ DNS サーバーを指定します。
      - セカンダリ **DNS**: 名前解決に使用するセカンダリ DNS サーバーを指定します。
      - 検索ドメイン: ホスト名を解決するときに使用する検索ドメイン名を指定します。たとえば、FQDN が console10.searchdomain.company.com の場合は、searchdomain.company.com と入力します。
      - **IPv4** アドレス: ホスト名にマッピングされる IP アドレス。
      - **IPv4** サブネット マスク: IPv4 アドレスのサブネット マスク。
      - **IPv4** ゲートウェイ アドレス: IPv4 アドレスのゲートウェイ アドレス。
12. *次へ*を選択します。
13. *完了準備完了*ウィンドウで詳細を確認し、*完了*を選択します。

    vSphere タスク バーには、コンソール エージェントの展開の進行状況が表示されます。
14. VMの電源をオンにします。

デプロイメントが失敗した場合は、エージェント ホストから検証レポートとログにアクセスできます。"インストールの問題をトラブルシューティングする方法を学びます。"

NetApp Consoleにコンソールエージェントを登録する

コンソールにログインし、コンソール エージェントを組織に関連付けます。ログイン方法は、コンソールを使用しているモードによって異なります。コンソールを標準モードで使用している場合は、SaaS Web サイトからログインします。コンソールを制限モードまたはプライベート モードで使用している場合は、コンソール エージェント ホストからローカルにログインします。

手順

1. Web ブラウザを開き、コンソール エージェント ホストの URL を入力します。

   コンソール ホスト URL は、ホストの構成に応じて、ローカルホスト、プライベート IP アドレス、またはパブリック IP アドレスになります。たとえば、コンソール エージェントがパブリック IP アドレスのないパブリック クラウドにある場合は、コンソール エージェント ホストに接続しているホストのプライベート IP アドレスを入力する必要があります。

2. サインアップまたはログインしてください。
3. ログイン後、コンソールを設定します。
   a. コンソール エージェントに関連付けるコンソール組織を指定します。
   b. システムの名前を入力します。
   c. 安全な環境で実行していますか**?** の下で、制限モードを無効のままにします。

      コンソール エージェントがオンプレミスにインストールされている場合、制限モードはサポートされません。

   d. *始めましょう*を選択します。

コンソールにクラウドプロバイダーの資格情報を追加する

コンソール エージェントをインストールしてセットアップしたら、コンソール エージェントが AWS または Azure でアクションを実行するために必要な権限を持つように、クラウド資格情報を追加します。

AWS

開始する前に

これらの AWS 認証情報を作成したばかりの場合は、使用可能になるまでに数分かかることがあります。資格情報をコンソールに追加する前に、数分待ってください。

手順

1. *管理 > 資格情報*を選択します。
2. *組織の資格情報*を選択します。
3. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: *Amazon Web Services > エージェントを選択します。
   b. 資格情報の定義: AWS アクセスキーとシークレットキーを入力します。
   c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。
   d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

これで、 "NetApp Console"コンソール エージェントの使用を開始します。

Azure

開始する前に

これらの Azure 資格情報を作成したばかりの場合は、使用可能になるまでに数分かかることがあります。コンソール エージェントに資格情報を追加する前に、数分間お待ちください。

手順

1. *管理 > 資格情報*を選択します。
2. *資格情報の追加*を選択し、ウィザードの手順に従います。
   a. 資格情報の場所: **Microsoft Azure >** エージェント を選択します。
   b. 資格情報の定義: 必要な権限を付与する Microsoft Entra サービス プリンシパルに関する情報を入力します。
      - アプリケーション（クライアント）ID
      - ディレクトリ（テナント）ID
      - クライアントシークレット
   c. マーケットプレイス サブスクリプション: 今すぐサブスクライブするか、既存のサブスクリプションを選択して、マーケットプレイス サブスクリプションをこれらの資格情報に関連付けます。
   d. 確認: 新しい資格情報の詳細を確認し、[追加] を選択します。

結果

これで、コンソール エージェントに、ユーザーに代わって Azure でアクションを実行するために必要なアクセス許可が付与されました。これで、 "NetApp Console"コンソール エージェントの使用を開始します。

オンプレミスのコンソールエージェントのポート

コンソール エージェントは、オンプレミスの Linux ホストに手動でインストールされる場合、受信 ポートを使用します。計画の際にはこれらのポートを参照してください。

これらの受信ルールは、すべてのNetApp Console展開モードに適用されます。

プロトコル	ポート	目的
HTTP	80	• クライアントのWebブラウザからローカルユーザーインターフェースへのHTTPアクセスを提供します • Cloud Volumes ONTAPのアップグレードプロセス中に使用されます
HTTPS	443	クライアントのWebブラウザからローカルユーザーインターフェースへのHTTPSアクセスを提供します