ORACLE
Cloud Infrastructure

Oracle Key VaultとのOracle Cloud Infrastructure のOracle Database暗号化キーの管理

Oracle Key Vault内のOracle Advanced Security TDE鍵の保護

ORACLE WHITEPAPER | 2018年5月

ORACLE

免責事項

下記事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。マテリアルやコード、機能の提供をコミットメント（確約）するものではなく、購買を決定する際の判断材料になさらないで下さい。オラクルの製品に関して記載されている機能の開発、リリース、および時期については、弊社の裁量により決定されます。

注意: このホワイト・ペーパーは、さらに改訂される場合があります。所有しているバージョンが最新であることを確認してください。

改訂履歴

このホワイト・ペーパーは、初版の公開後、次の改訂がありました。

日付	改訂内容
2018年5月25日	• Oracle Cloud Infrastructure環境でOracle Key Vaultを管理するための運用上のベスト・プラクティスを追加 • FAQの項を削除

Oracle Cloud Infrastructureホワイト・ペーパーの最新のバージョンは、https://cloud.oracle.com/iaas/technical-resourcesで入手できます。

目次

はじめに

Oracle Databaseは、Oracle Advanced Securityの一部として提供されている、透過的データ暗号化(TDE)を使用した保存データの暗号化を実装しています。TDEは、データベースを、そのデータを使用するより高レベルのアプリケーションに対して透過的に暗号化するものであり、表領域全体または表内の特定の列に実装可能です。

TDEは、表領域または列の暗号化に使用されるデータベース暗号化鍵およびデータベース暗号化鍵のラップ(暗号化)に使用されるTDEマスター鍵で構成される2層の暗号化鍵アーキテクチャを使用します。ラップされるデータベース表領域および列の暗号化鍵は、データベースに元々格納されており、TDEマスター鍵は、ローカル・ファイルシステム、またはクラスタ化されたアクセスのための自動ストレージ管理(ASM)ディスク・グループ内のOracleウォレットに通常格納されます。TDEマスター鍵を格納するその他の推奨オプションとして、Oracle Key Vaultまたはハードウェア・セキュリティ・モジュール(HSM)などの集中化された鍵管理プラットフォームがあります。

TDEマスター鍵は、OracleウォレットにPKCS#12形式のファイルで格納され、顧客が指定するパスワードで保護されます。データベース暗号化鍵がデータベースの暗号化または復号化に必要な場合、顧客は、正しいパスワードを指定してウォレットを開き、ウォレットベースのTDEマスター鍵が、データベース暗号化鍵のアンラップに使用されます。続いて、このデータベース暗号化鍵は、データベース表または列の暗号化または復号化のためにデータベースによって使用されます。完全自動の運用要件のために、Oracleウォレットは、自動ログイン・オプションを使用して、パスワードなしでプロビジョニングすることもできます。

仮想マシン(VM)とは異なり、Oracle Cloud Infrastructure Computeベア・メタル・インスタンスは、Oracle Cloud Infrastructureによって制御される高特権のハイパーバイザによって管理されず、Oracle Cloud Infrastructureのオペレータがベア・メタル・インスタンスのメモリーまたはローカル・ディスクに格納されているデータにアクセスすることは技術的に実行不可能です。このため、顧客はベア・メタル・インスタンスに格納されているデータを完全に制御できます。Oracle Cloud Infrastructureの顧客に提供されるOracle Database機能は、ベア・メタル・インスタンスを活用しています。Oracle Cloud Infrastructure内のOracle Databaseオプションは次のとおりです。

- **Databaseインスタンス:** NVMeローカル・フラッシュ・ストレージを備えた、ベア・メタル・インスタンス上の柔軟でオンデマンドのOracle Database。次のデータベース・インスタンスのシェイプが提供されています。
  - **HighIO:** 36コア、512GB RAMおよび12.8TB NVMeストレージ
  - **DenseIO:** 36コア、512GB RAMおよび28.8TB NVMeストレージ
  - **2ノードRAC**
  - **Exadata:** クォータ・ラック、ハーフ・ラックおよびフル・ラック
- **BYOL (Bring your own license)データベース:** 顧客は、自身のベア・メタル・インスタンスにOracle Databaseをインストールできます。

DatabaseおよびBYOLベア・メタル・インスタンスでは、TDEマスター鍵は、顧客が所有するベア・メタル・インスタンス上のOracleウォレットで通常プロビジョニングされるため、顧客はTDE鍵全体を完全に制御できます(つまり、Oracle Cloud Infrastructureオペレータは、ベア・メタル・インスタンス上のOracleウォレットにアクセスできません)。この場合、Oracle Cloud Infrastructureの顧客は、所有しているすべてのデータベース・インスタンスでOracleウォレットのTDEマスター鍵を個別に管理する責任があります。結果として発生する鍵管理は、データベース・デプロイメントが大規模な一部の顧客に、簡単ではない運用面の労力を強いる場合があります。

Oracle Key Vaultは、TDE鍵管理に関連する運用面の労力を軽減するソリューションを提供します。Oracle Key Vaultは、複数のOracleデータベース、およびMySQL TDE、Solaris CryptoおよびASM Cluster File System (ACFS)暗号化など、その他のセキュリティ・アプリケーションのTDEマスター鍵を格納し管理するために使用される、セキュリティが強化されたソフトウェア・アプライアンスです。顧客は、自身のベア・メタル・インスタンスにOracle Key Vaultをインストールし、すべてのTDEマスター鍵を集中的に管理するために、Oracle Cloud InfrastructureのすべてのOracleデータベースをエンドポイントとして登録できます。Oracle Key Vaultを自身のベア・メタル・インスタンスにインストールすることで、Oracle Cloud Infrastructureの顧客は、Oracle Key Vaultの機能により鍵管理の運用面の労力を軽減しながら、TDEマスター鍵全体の制御を引き続き維持できます。

このホワイト・ペーパーでは、仮想クラウド・ネットワーク(VCN)内の顧客所有のベア・メタル・インスタンスにOracle Key Vaultをインストールして構成し、OracleデータベースのTDE鍵を管理する手順を示します。

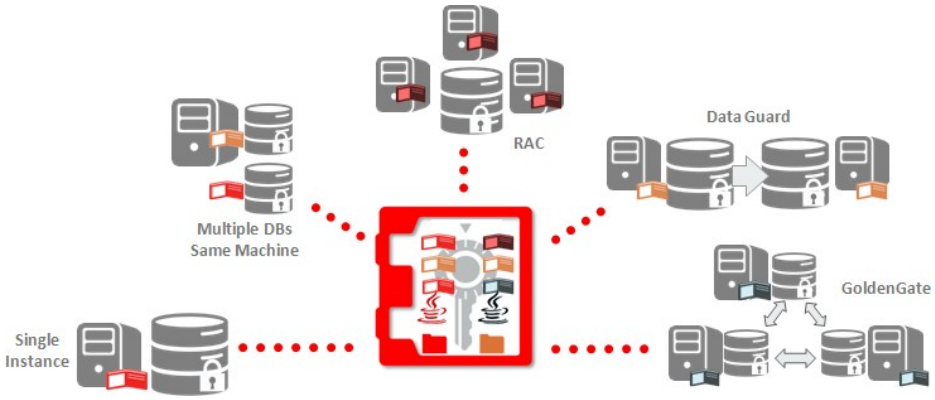
Oracle Key Vaultの概要

Oracle Key Vaultにより、暗号化鍵、Oracleウォレット、Javaキーストアおよび資格証明ファイルを集中的に管理することで、暗号化およびその他のセキュリティ・ソリューションを迅速に導入できます。Oracle Key Vaultは、Oracle Advanced SecurityのTransparent Data Encryption (TDE)マスター鍵の管理用に最適化されています。このセキュリティが強化されたフルスタックのソフトウェア・アプライアンスは、セキュリティ、可用性およびスケーラビリティを得るためにOracle LinuxおよびOracle Databaseテクノロジを使用しています。Oracle Key Vaultは、OASIS Key Management Interoperability Protocol (KMIP)業界標準をサポートしています。

Oracleウォレットおよびキーストアの集中管理

OracleウォレットおよびJavaキーストアは、多くの場合、サーバーおよびサーバー・クラスタ全体に手動で分散されます。Oracle Key Vaultは、これらのファイルの内容を項目化してマスター・リポジトリ内に格納し、同時に、サーバー・エンドポイントがそのローカル・コピーを使用してOracle Key Vaultから切断された状態で動作し続けることを可能にします。ウォレットおよびキーストアは、アーカイブ後、ローカル・コピーを誤って削除した場合や、パスワードを忘れた場合に、サーバーにリカバリできます。

Oracle Key Vaultは、Oracle RAC、Oracle Active Data GuardおよびOracle GoldenGateなどのデータベース・クラスタ全体でウォレットの共有を合理化します。ウォレット共有の保護は、Oracle Data PumpおよびOracle Transportable Tablespacesを使用した暗号化済データの移動も促進します。Oracle Key Vaultは、Oracle MiddlewareおよびOracle Databaseのサポートされているすべてのリリースから、Oracleウォレットとともに使用できます。

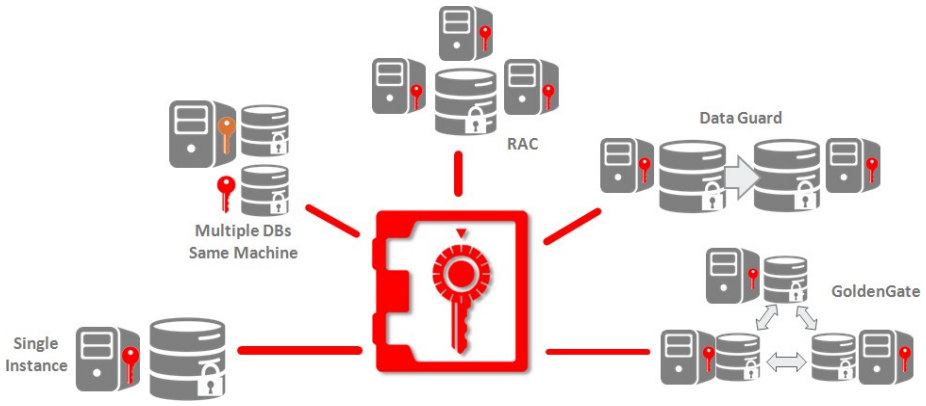


Oracle Key Vaultのウォレット管理のシナリオ

オンラインによるTDEマスター鍵管理

TDEを使用するOracleデータベースの場合、Oracle Key Vaultは、ローカル・ウォレット・ファイルの使用の代替として、直接ネットワーク接続でTDEマスター鍵を集中的に管理します。この接続により、定期的なパスワード・ローテーション、ウォレット・ファイルのバックアップおよび忘れられたパスワードのリカバリなどのウォレット・ファイル管理の運用面の課題が解消されます。また、この方法は、法規制の順守でたびたび言及される暗号化鍵と暗号化済データの物理的な分離も提供します。Oracle Key Vaultに格納されるマスター鍵は、エンドポイントのアクセス制御設定に従って、データベース全体の表領域鍵または表鍵の復号化に使用可能です。

このローカル・ウォレット・コピーを使用しない鍵の共有方法は、TDEがOracle RAC、Oracle Active Data GuardおよびOracle GoldenGateなどのデータベース・クラスタで実行されている場合に便利です。Oracleデータベース内の暗号化済データに使用される既存のマスター鍵は、初期設定の一環としてOracleウォレットからOracle Key Vaultに簡単に移行できます。TDEとOracle Key Vaultの直接ネットワーク接続は、データベースへのパッチ適用の必要なく、Oracle Database 11gR2とOracle Database 12cでサポートされています。



Oracle Key VaultのオンラインTDEマスター鍵シナリオ

資格証明ファイルの集中バックアップ

SSH鍵、Kerberos keytabファイルを含む資格証明ファイルおよび類似の資格証明ファイルも、適切な保護メカニズムなしで広範囲に分散されています。Oracle Key Vaultは、長期間にわたる保持とリカバリのために資格証明ファイルをバックアップします。Oracle Key Vaultは、必要な場合、このファイルを簡単にリカバリし、このファイルへのアクセスを監査し、信頼できるエンドポイント全体で共有します。

加えて、Oracle Key Vaultは、MySQL Transparent Data Encryption、Solaris CryptoおよびASM Cluster File System (ACFS)ファイル暗号化ソリューションの鍵管理を集中化します。

Oracle Cloud InfrastructureへのOracle Key Vaultのインストール

Oracle Key Vaultは、オンプレミス・ネットワークの物理ホスト上にソフトウェア・アプライアンスとしてインストールするように設計されています。これらのデプロイメント機能により、Oracle Key Vaultの現在バージョンは、ベア・メタル・インスタンスにそのままではインストールできませんが、そのかわり、顧客所有のベア・メタル・インスタンスにVMとしてインストールされます。顧客は、Oracle Key Vault VMをインストールする前に、ベア・メタル・インスタンスにハイパーバイザをインストールします。このBYOH (bring-your-own-hypervisor)モデルでは、顧客はハイパーバイザを管理者として管理し、ベア・メタル・インスタンスおよびそこで実行しているOracle Key Vault VMの完全な制御を許可します。

この項では、Oracle Key Vaultのイメージとライセンスの取得、ハイパーバイザのインストール、およびベア・メタル・インスタンスへのOracle Key Vault VMのインストールに関する情報を提供します。次の説明では、KVMハイパーバイザを使用します。

Oracle Key Vaultのイメージおよびライセンスの取得

ダウンロードの説明に従って、インストールするOracle Key Vault ISOイメージをダウンロードします。インストールおよび必要な管理タスクについては、ドキュメントを参照してください。

Oracle Key Vaultは、Oracle Database Security製品ポートフォリオ内で個別にライセンスされる製品です。すべての本番および非本番(テストおよび開発)環境で必要なライセンスを入手してください。

BYOH KVMのインストールおよびOracle Key Vault VMのためのVCNネットワーキングの構成

BYOHの場合、必須の機能はVCNのセカンダリVNICです。セカンダリVNICにより、追加のVNICをベア・メタル・インスタンスにアタッチし、VCNルーティング可能なIPアドレスをVNICに割り当て、BYOHベア・メタル・インスタンスで実行されているVMにアタッチできます。セカンダリVNICの詳細は、ネットワーキング・サービスのドキュメントを参照してください。

この項では、完全を期すためにBYOH KVMの大まかな手順を紹介します。詳細な手順は、対応する『Installing and Configuring KVM on Bare Metal Instances with Multi-VNIC』ホワイト・ペーパーを参照してください。大まかな手順は次のとおりです。

1. Oracle Linux 7.*x*イメージを使用してベア・メタル・インスタンスを起動します。

2. SSH鍵を使用してベア・メタル・インスタンスにログインし、接続をテストします。接続できない場合、VCNセキュリティ・リストおよびファイア・ウォール・ルールを確認してください。VNCクライアントを使用してインスタンスに接続できるようにするには、ベア・メタル・インスタンス上にVNCサーバーをインストールすることをお薦めします。Oracle Linux上にVNCサーバーを構成する手順は、https://docs.oracle.com/cd/E52668_01/E54669/html/ol7-vnc-config.htmlで参照できます。

3. Oracle Cloud Infrastructureコンソールで、最小256GBのブロック・ストレージ・ボリュームを作成し、ベア・メタル・インスタンスにアタッチします。アタッチされたボリュームにファイルシステムをマウントし、マウントしたシステムにOracle Key Vault ISOをコピーします。格納されるファイル数に応じて、1TBのブロック・ボリュームの使用をお薦めします。

4. コンソールまたはAPIを使用して、セカンダリVNICをアタッチし、セカンダリ・インタフェースのIPアドレス、MACアドレスおよびVLANタグを書き留めます。これは、Oracle Cloud Infrastructure Databaseインスタンスなどの他のVCNホストからのネットワーク・アクセスが可能なように、Oracle Key Vault VMに割り当てられるセカンダリIPアドレスであることに注意してください。

5. KVMハイパーバイザをベア・メタル・インスタンスにインストールします。

```
sudo yum install qemu-kvm qemu-img virt-manager libvirt libvirt-python
libvirt-client virt-install virt-viewer bridge-utils
```

6. SR-IOVを有効にし、ベア・メタル・インスタンスを再起動します。詳細は付録を参照してください。

7. ベア・メタル・インスタンスの起動後、OSのSR-IOV仮想機能(VF)を有効にします。VFを選択し、前に作成したセカンダリVNICのMACアドレスで構成します。詳細は付録を参照してください。

8. セカンダリVNICのVLANタグを使用して、ネットワーク・インタフェースを作成します。インタフェースは、前のステップで構成したVFとブリッジ接続されます。詳細は付録を参照してください。

9. ベア・メタル・インスタンス上でpifconfigを実行して、作成されたネットワーク・デバイスを表示します。

ベア・メタル・インスタンスへのOracle Key Vault VMのインストール

1. `qemu-img`を使用して、500G仮想ディスクを作成します。この仮想ディスクは、Oracle Key Vault VMによって使用されます。

```
qemu-img create -f raw <path_to_disk_image> 500G
```

2. `virt-install`を使用して、Oracle Key Vault VMをインストールします。

```
sudo virt-install --arch=x86_64 --name=<OKV_VM_name> --ram 16000 --cpu
Haswell-noTSX --vcpus=4 --hvm --video qxl --nonetwork --os-type linux --
noautoconsole --boot hd,cdrom -disk
<path_to_OKV_ISO>,device=cdrom,bus=ide
-disk <path to OKV VM disk image>,format=raw,bus=scsi --graphics
```

前述のコマンドは、ブート・ログを表示するためにOracle Key Vault VMコンソールへのVNC接続も作成します。

3. ローカルホストにSSHトンネルを作成し、VNCクライアントを使用してOracle Key Vault VMコンソールに接続します。これは、インストールでエラーが発生した場合に特に便利です。

```
ssh -i <bare_metal_SSH_key> -L <VNC_port>:localhost:<VNC_port>
opc@<bare_metal_host_IP>
```

`<bare_metal_SSH_key>`は、ベア・メタル・インスタンスに接続するためのSSH鍵であり、`<VNC_port>`は、ステップ2の`virt-install`で指定されたポート番号であり、`<bare_metal_host_IP>`は、ベア・メタル・インスタンスのIPアドレスです。

Macでは、ネイティブVNCクライアント(画面共有)を使用して、ステップ2で構成した`vnc://opc@localhost:<VNC_port>`および`<VNC_password>`により、Oracle Key Vault VMコンソールに接続できます。

4. virshを使用して、VNICネットワーク・インタフェース(前の項で作成)をアタッチします。正しいVNIC MACアドレスおよびネットワーク・デバイス名を`attach.xml`ファイルに保存する必要があります(ファイルの詳細は付録を参照してください)。VNICネットワーク・インタフェースをアタッチした後、Oracle Key Vault VMを破棄して再起動します。

```
sudo virsh attach-device <VM_name> ./attach.xml -config

sudo virsh destroy <VM_name>

sudo virsh start <VM_name>
```

VMはインストールを開始すると、VMにアタッチされているVNICネットワーク・デバイスを検出します。VMのインストールは約30分かかります。Oracle Key Vaultのインストールの詳細は、https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_install.htm#OKVAG10641で入手できます。

インストール中、次の情報を求められます。

- **Oracle Key Vaultインストール・パスフレーズ:** このパスフレーズは、Oracle Key Vaultコンソールへの初回ログインに使用されます。
- **Oracle Key Vaultネットワーク構成:** これには、Oracle Key Vault VM IPアドレス、ゲートウェイIPアドレスおよびネットマスクが含まれます。アタッチされているセカンダリVNICをOracle Key Vault VM (OKV_VM_IP) IPアドレスとして指定します。ゲートウェイIPアドレスに10.0.0.1、およびネットマスクとして255.255.255.0を指定します。

5. インストールの完了後、ホスト・ベア・メタル・インスタンスでWebブラウザを開き、`https://OKV_VM_IP`を入力します(`OKV_VM_IP`はOracle Key Vault VMのIPアドレスです)。

   ブラウザでOracle Key Vaultコンソールが開きます。
6. インストール・パスフレーズを使用してログインします。
7. プロンプトが表示されたら、鍵管理者、システム管理者および監査マネージャのユーザー名およびパスワードを設定します。また、プロンプトが表示されたら、リカバリ・パスワード(セキュア・バックアップからの鍵のリカバリに使用)、ルート・パスワード(VM上のルート権限)およびサポート・パスワード(VMへのSSHアクセス用)を設定します。Oracle Key Vaultコンソールで作成するユーザーの詳細は、https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_install.htm#OKVAG10740で入手できます。
8. RESTサービスがOracle Key Vaultで有効であることを確認します。Oracle Key Vaultコンソールで「**システム**」の下の「**RESTfulサービス**」チェック・ボックスを選択し、構成を保存します。すべてのエンドポイントが登録されプロビジョニングされた後、RESTサービスを無効にすることもできます。

Oracle Cloud InfrastructureのOracle Key VaultによるOracle TDE鍵の構成

この項では、OracleデータベースのOracle Key Vaultエンドポイントとしての登録、およびOracleウォレットからOracle Key VaultへのTDEマスター鍵の移行の手順を説明します。これらのタスクはOracle Key Vault RESTfulユーティリティを使用して実行します。RESTfulユーティリティは、データベース・インスタンスで実行され、データベースをエンドポイントとしてOracle Key Vault KMIPサーバーとして登録するKMIPクライアントです。

複数のデータベース・インスタンスを登録する場合、Oracle Key Vault RESTfulユーティリティを使用して、プログラムによって登録プロセスを自動化できます。

Oracle Key Vaultへのデータベース・エンドポイントの登録

1. `https://HOST_BARE_METAL_IP`を使用して、データベース・インスタンスからOracle Key Vaultコンソールにログインします。前の項のIP表ルールによって、ホスト・ベア・メタル・インスタンス上のポート443がOracle Key Vault VMに転送されます。

2. コンソールで、「**RESTfulサービス・ユーティリティ**」をクリックして、ユーティリティをインスタンスにダウンロードします。ユーティリティ(`okvrestservices.jar`)は`/home/oracle/Downloads`(`/home/oracle`はデータベース・インスタンスのホーム・ディレクトリです)にダウンロードされたものとします。

3. `Downloads`ディレクトリで、次の2つのファイルを作成します(これらのファイルの代表的な例については、「付録」を参照してください)。
   - `rest.ini`構成ファイル
   - `enroll_okv_endpoints`スクリプト・ファイル

4. データベース・インスタンスの`home`ディレクトリで、`okvhome`ディレクトリを作成します(`/home/oracle/okvhome`)。

5. 次のコマンドを使用して、データベースをエンドポイントとして登録して、TDEマスター鍵を格納するOracle Key Vault仮想ウォレットを作成します。

   ```
   java -jar okvrestservices.jar --config rest.ini --script
   enroll_okv_endpoint
   ```

   コマンドにより、Oracle Key Vault管理パスワードの入力を求められます。

   `CUSTOMER_DB`がOracleデータベース・エンドポイントの名前、`CUSTOMER_DB_WALLET`がデータベースTDEマスター鍵を保持するOracle Key Vault仮想ウォレットと仮定します。コマンドが完了すると、`CUSTOMER_DB`がOracle Key Vaultコンソールの「エンドポイント」タブにリストされているはずです。「キーとウォレット」タブに`CUSTOMER_DB_WALLET`という仮想ウォレットも表示されているはずです。この時点では、`CUSTOMER_DB_WALLET`は空です。コマンドは、`oracle/okvhome`ディレクトリ内に`CUSTOMER_DB`サブディレクトリも作成し、これにはさまざまなユーティリティ(次の項で使用するokvutilなど)が含まれます。

6. `/home/oracle/okvhome/CUSTOMER_DB/bin/root.sh`をroot (またはsudo)として実行します。これにより、Oracle Key Vault PKCS#11ドライバが、指定したOracleデータベース・ファイルシステム内の場所にコピーされます(`/opt/oracle/extapi/64/hsm/oracle/1.0.0`)。Oracleデータベースは、このドライバ内の機能を使用して、Oracle Key Vaultと相互にやりとりします。

Oracle Databases 12.1.0.2および12.2.0.1では、Oracle Key Vaultウォレットが各コンテナ・データベース(CDB)に作成され、CDBのすべてのマルチテナント・プラガブル・データベース(PDB)のすべてのTDEマスター鍵は同じウォレットに格納されます。これは、実行中のPDBの数に関係ありません。これは、個別のOracle Key VaultウォレットをPDBに作成できる将来のバージョンでは変更される可能性があります。

Oracle Key VaultへのOracleウォレットの鍵のアップロード

次のコマンドを使用して、Oracleウォレットの内容をOracle Key Vaultにアップロードします。たとえば、Oracleウォレットは、Oracleデータベース・インスタンス上の`/etc/oracle/wallets/orcl`に配置されます(orclはデータベースSID名です)。自分のデータベースSIDと置き換えます。

```
/home/oracle/okvhome/CUSTOMER_DB/bin/okvutil upload -t WALLET -g
CUSTOMER DB WALLET -l /etc/oracle/wallets/orcl
```

コマンドにより、Oracleウォレットのパスワードの入力を求められます。

このコマンドが完了すると、Oracle Key Vaultコンソールの「**キーとウォレット**」タブにさまざまな鍵および証明書識別子が表示されます。

また、次のコマンドを使用して、Oracle Key Vault内の`CUSTOMER_DB`エンドポイントの鍵および証明書のIDをリストすることもできます。結果は、Oracle Key Vaultコンソール内の`CUSTOMER_DB`の下にリストされた内容に対応しているはずです。

```
/home/oracle/okvhome/CUSTOMER_DB/bin/okvutil list
```

OracleウォレットからOracle Key VaultへのTDEマスターの移行

1. `sqlplus`ターミナル内の次のコマンドを使用して、データベース・インスタンスのOracleウォレットを閉じます。`WALLET_PASSWD`は、データベース・インスタンスのOracleウォレットのパスワードです。

   ```
   administer key management set keystore close identified by
   “WALLET_PASSWD”;
   ```

2. `$ORACLE_HOME/network/admin/sqlnet.ora`構成ファイルを`(METHOD=FILE)`から`(METHOD=HSM)`に変更します。

3. 次のコマンドを使用して、変更が有効であることを確認します。`sqlplus / as sysdba`を使用してデータベースにログインし、`sqlplus`ターミナルでコマンドを発行します。`FILE`と`HSM`の両方が`CLOSED`である必要があります。

   ```
   select wrl_type,status from v$encryption_wallet;
   ```

4. sqlplusターミナルで次のコマンドを使用して、OracleウォレットからOracle Key VaultにTDEマスター鍵を移行します。コマンド内の"null"は、データベース・エンドポイントのnullパスワードを示しています。(null以外の値を使用する場合、データベースがOracle Key VaultのTDEマスター鍵にアクセスする際は必ずパスワードを求めるプロンプトが表示され、その結果、完全自動の運用に問題が発生する可能性があります。)

```
administer key management set encryption key identified by “null”
migrate using “WALLET PASSWD” with backup;
```

移行されたTDEマスター鍵識別子は、Oracle Key Vaultコンソールの「**キーとウォレット**」タブに表示されます。

5. その後、sqlplusターミナルで次のコマンドを使用して、Oracle Key Vault内のTDEマスター鍵をローテーションできます。

```
administer key management set encryption key identified by “null”;
```

ヒント: TDEマスター鍵をOracleウォレットに戻すには、$ORACLE_HOME/network/admin/sqlnet.ora構成ファイルで(METHOD=HSM)を(METHOD=FILE)に変更し、sqlplusターミナルで次のコマンドを発行します。

```
administer key management set encryption key identified by “null”
reverse migrate using “WALLET PASSWD” with backup;
```

Oracle Key Vaultのベスト・プラクティス

次のOracle Key Vaultベスト・プラクティスを使用して、セキュリティおよびオペレーションを強化してください。

非クリティカルなデータベース・ワークロードを使用したプロトタイプの作成

TDEマスター鍵を失うと、Oracleデータベース内の暗号化済データ(暗号化された表領域および列)にアクセスできなくなります。したがって、Oracle Key Vaultなどの鍵管理ソリューションを導入する際は、適切でかつ非クリティカルな開発またはテスト・データベースのワークロードを使用して、鍵管理ソリューションのプロトタイプの作成を最初に行うことを強くお薦めします。このプランにより、ミッションクリティカルな本番ワークロードの処理に適した鍵管理アーキテクチャを形成できると同時に、鍵管理ソリューションのすべての側面を十分に理解できます。

鍵のバックアップの保護

障害回復(DR)をサポートするために、格納しているOracle Key VaultのTDEマスター鍵のセキュア・バックアップを定期的に行う必要があります。TDEマスター鍵を失うと、暗号化されたデータベース・データにアクセスできなくなります。次に、Oracle Key VaultのTDEマスターのバックアップの作成オプションを示します。

- **自動ブロック・ボリューム・バックアップ:** Oracle Key Vault VMを格納するブロック・ストレージ・ボリュームの定期的なスナップショットを作成します。Oracle Cloud Infrastructureコンソールでバックアップおよび頻度を構成できます。Oracle Key Vaultアプライアンスは、保存されている鍵を元々暗号化するものであり、Oracle Cloud Infrastructureブロック・ストレージ・ボリュームは、Oracle Cloud Infrastructureコントロール・プレーンによって保存時に暗号化されることに留意してください。
- **自動Oracle Key Vaultのセキュア・バックアップ:** ホスト・ベア・メタル・インスタンスへのOracle Key Vaultの鍵の定期的な自動バックアップを構成します(ホスト・ベア・メタル・インスタンスのみがOracle Key Vault VMからアクセス可能であるためです)。加えて、ホスト・ベア・メタル・インスタンスのスペースを解放するために、Oracle Key Vaultの鍵のバックアップ・ファイルを顧客所有のOracle Cloud Infrastructure Object Storage内のバケットにコピーすることをお薦めします。DRのために、Oracle Key Vaultは、これらのバックアップからリカバリ・パスワードを使用してリストアされます。(Oracle Key Vaultのリカバリ・パスワードをなくさないようにしてください)。

  Oracle Key Vaultのセキュア・バックアップの手順は、次の場所から入手できます。
  https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_ha_backup.htm#OKVAG10746

高可用性のための構成

高可用性(HA)は、プライマリおよびスタンバイのOracle Key Vaultサーバーで構成され、鍵のオンライン・バックアップがプライマリからセカンダリに行われます。2つの異なるBYOHベア・メタル・ホストにプライマリとセカンダリのOracle Key Vault VMを保有することをお薦めします。

Oracle Key Vault SSHアクセスの有効化

SSHアクセスは、Oracle Key Vault VMのトラブルシューティングおよび運用アクティビティを行うのに便利であるため、VMへのSSHアクセスを有効にすることをお薦めします。Oracle Key Vaultコンソールで、システム設定に移動し、ホスト・ベア・メタル・インスタンスからOracle Key Vault VMへのSSHアクセスを有効にします。このステップの後、`ssh support@OKV_VM_IP`を使用して、ホスト・ベア・メタル・インスタンスからOracle Key Vault VMにログインできます。

SSHアクセスを有効にする手順は、次の場所で入手できます。
http://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_appliance.htm#OKVAG10885

Oracle Key Vault監査ログの使用

Oracle Key Vaultは、格納されているTDEマスター鍵に関連するすべてのアクションを監査証跡に記録します。監査証跡の各行には、誰(Oracle Key Vaultユーザー)がどのアクション(操作)をどのオブジェクト(TDEマスター鍵)に行ったか、およびそのアクションの結果が表示されます。このOracle Key Vaultの監査証跡は、監査のためにOracle Key Vault監査マネージャによってCSVファイルとしてエクスポートできます。

Oracle Key Vault監査証跡のエクスポートの手順は、次の場所で入手できます。
https://docs.oracle.com/cd/E65319_01/OKVAG/okv_appliance.htm#OKVAG10870

VCNセキュリティ・リストを使用したOracle Key Vaultインスタンスの保護

ホスト・ベア・メタル・インスタンスでVCNセキュリティ・リストを使用して、VCN内の認可済データベース・インスタンスからのみOracle Key Vault VMへのネットワーク接続を許可するように設定できます。セキュリティ・リストは、Oracle Key Vaultエンドポイントとして構成されているデータベース・インスタンスに対応したIPアドレスからの、ホスト・ベア・メタル・インスタンスのポート5696 (Oracle Key Vault KMIPサーバー・ポート)でのTCP接続のみを許可します。Oracle Key Vault Webコンソールにリモート・アクセスが必要な場合、ポート443でのアクセスも許可する必要があります。

Oracle Cloud Infrastructure環境でのOracle Key Vaultの管理

Oracle Key Vaultインスタンスは、これを使用してOracle Cloud InfrastructureデータベースのTDEマスター鍵を格納している顧客によって完全に管理されます。顧客は、そのVCN内のベア・メタル・インスタンス上のOracle Key Vaultインスタンスのインストールおよび構成、ならびにOracle Key Vaultエンドポイントとしてのデータベース・インスタンスの追加など、その管理の責任を負います。また、顧客は、格納されているOracle Key Vaultの鍵の自動バックアップの設定およびDRからのバックアップのリストアの責任も負います。Oracle Cloud Infrastructureは、Oracle Key Vault VMの管理に関与しません。

Oracle Cloud Infrastructure内で顧客によって管理されているOracleデータベースの場合、Oracle Key Vaultは、オンプレミスの場合と同様にOracle Cloud Infrastructure内で機能します。Oracle Cloud Infrastructureデータベース・インスタンスの場合、RMANおよびData Guardなどの特定の機能は自動化され、TDE鍵はデータベース・インスタンス上のOracleウォレット内に存在していると見なします。Oracle Key VaultをOracle Cloud Infrastructureデータベース・インスタンスとともに使用する必要がある場合、Oracle Cloud Infrastructureチームにご相談ください。

Oracle Key VaultのActive Data Guardの構成

Oracle Key Vaultを使用すれば、Active Data Guard構成内のプライマリ・データベースとスタンバイ・データベース間で鍵を手動でコピーする手順が不要になります。これは、プライマリ・データベースとスタンバイ・データベース間で共有Oracle Key Vaultウォレットを登録することで可能になります。

手順については、次の場所にアクセスしてください。
https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_scenarios.htm#OKVAG10849

Oracle Key VaultのRACの構成

Oracle Key Vault仮想ウォレットを定義し、これをすべてのRACノード間で共有デフォルト・ウォレットとして構成します。

手順については、次の場所にアクセスしてください。
https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_scenarios.htm#OKVAG10704

Oracle Key VaultのGoldenGateの構成

ソースOracleデータベースがOracle Key Vaultエンドポイントとして構成されている場合、GoldenGateパスワードは、ソース・データベースのTDEマスター鍵と同じOracle Key Vault仮想ウォレットに格納されます。ターゲット・データベースは、別のOracle Key Vaultエンドポイントとして構成されます。

手順については、次の場所にアクセスしてください。
https://docs.oracle.com/cd/E50341_01/doc.1210/e41361/okv_scenarios.htm#OKVAG10845

結論

このホワイト・ペーパーでは、Oracle Cloud Infrastructure内のOracleデータベースのTDEマスター鍵を管理するためのOracle Key Vaultソリューションを紹介しました。Oracle Key Vaultアプライアンスは、顧客が所有するベア・メタル・インスタンス上で実行され、すべての鍵の完全な制御を可能にするだけではなく、Oracle Key Vaultの機能を活用することで、Oracle Cloud Infrastructure内の複数のOracleデータベースのTDEマスター鍵の管理に伴う運用面の労力を軽減します。運用要件に応じて、OracleウォレットとOracle Key Vaultを組み合せて使用し、Oracle Cloud Infrastructure内のOracleデータベース暗号化鍵を管理することもできます。

付録

ベア・メタル・インスタンスでのSR-IOVの有効化

1. `/etc/default/grub`ファイルの`GRUB_CMDLINE_LINUX`行に`intel_iommu=on`を追加します。
2. 新しいgrub構成ファイルを生成します。

```
grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

3. ベア・メタル・サーバーを再起動します。

VFの有効化およびセカンダリVNICのMACアドレスによる構成

1. 仮想機能(VF)を有効にし、`vepa`ブリッジング・モードを設定します。Oracle Linux上で、`ens2f0`は物理インタフェースです。

```
echo “16” > /sys/class/net/ens2f0/device/sriov_numvfs

bridge link set dev ens2f0 hwmode vepa
```

2. 使用可能なVFをリストします。使用可能なVFのVF番号(VF_NUM)をメモします。

```
ip link show ens2f0
```

3. VNICのMACアドレス(VNIC_MAC)を使用してVFを構成します。

```
ip link set ens2f0 vf VF_NUM mac VNIC_MAC spoofchk off
```

セカンダリVNICのVLANタグを使用した、ネットワーク・インタフェースの作成

1. VFネットワーク・デバイス名(VF_DEVICE_NAME)を取得します。

   VF番号が付けられたVF_NUMについて、次のコマンドの出力で(VF_NUM+1)行番号を選択します。たとえば、VF_NUMが1の場合、出力の2行目を選択します。ポート、スロットおよびファンクション番号が、16進形式で行の最初のフィールドとしてリストされます。たとえば、13:10:2は、ポート番号19、スロット番号16およびファンクション番号2を意味し、VF_DEVICE_NAMEはenp19s16f2となります。

```
lspci -nn | grep -i virtual
```

2. VFネットワーク・デバイスを起動します。

```
ip link set VF_DEVICE_NAME down

ip link set VF_DEVICE_NAME up
```

3. VNIC VLANにVFネットワーク・デバイスを割り当てます。

```
ip link add link VF_DEVICE_NAME name VLAN_DEVICE_NAME type vlan id
VNIC_VLAN_TAG

ip link set VLAN_DEVICE_NAME up
```

attach.xmlファイル

```
<interface type='direct'>
    <mac address='<VNIC_MAC>'/>
    <source dev='<VLAN_DEVICE_NAME>' mode='passthrough'/>
    <model type='e1000'/>
</interface>
```

rest.iniファイル

`rest.ini`ファイルの内容は次のとおりです。

```
server=OKV_VM_IP
usr=<OKV_ADMIN_USER>
log_level=ALL
```

`<OKV_ADMIN_USER>`は、Oracle Key Vaultコンソールへのログインに使用されるユーザー名です。

enroll_okv_endpointファイル

```
create_wallet -wallet_name CUSTOMER_DB_WALLET
create_endpoint -ep_name CUSTOMER_DB -ep_platform LINUX64 -ep_type ORALE_DB
set_default_wallet -ep_name CUSTOMER_DB -wallet_name CUSTOMER_DB_WALLET
provision -autologin -ep name CUSTOMER DB -dir /home/oracle/okvhome
```

ファイル内の`CUSTOMER_DB`は、Oracleデータベース・エンドポイントの名前であり、`CUSTOMER_DB_WALLET`は、登録されているデータベースのTDEマスター鍵を含むOracle Key Vault仮想ウォレットです。エンドポイントとして登録されている各Oracleデータベースの場合、一意の名前をスクリプトで指定する必要があります。エンドポイントとして登録する予定の各データベースについては、データベースのエンドポイントおよびウォレット名に次の形式を使用することをお薦めします。

- `ORACLE_DB_$SID` (`$SID`はOracleデータベースのSIDです)
- `ORACLE_DB_$SID_WALLET` (`$SID`はOracleデータベースのSIDです)

ORACLE

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

Oracle Key VaultとのOracle Cloud InfrastructureのOracle Database暗号化キーの管理
2018年5月
著者: Nachiketh Potlapally、Saikat Saha