



Kaspersky Hybrid Cloud Security ~ Public Cloud Security

2021年3月13日

株式会社カスペルスキー

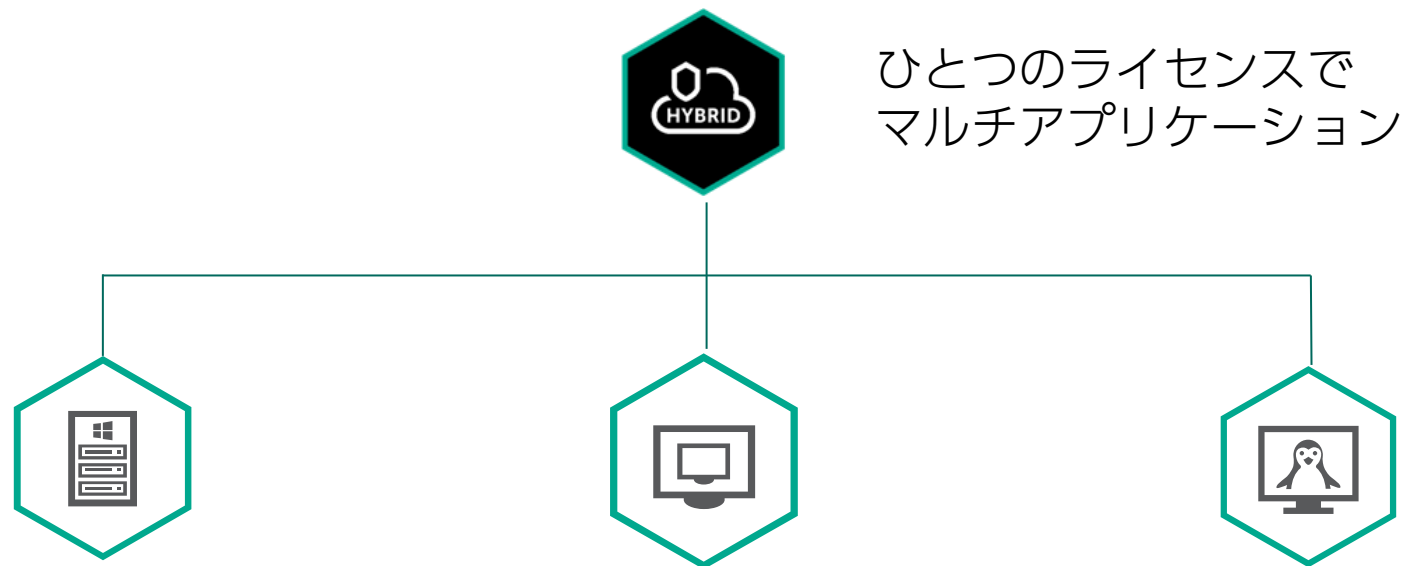
セールスエンジニアリング部

kaspersky

Kaspersky Hybrid Cloud Security

Kaspersky Hybrid Cloud Securityは、クラウド環境、仮想環境、物理サーバーに使用出来る製品です。

ライセンス種別に合わせて、保護範囲が決まります。



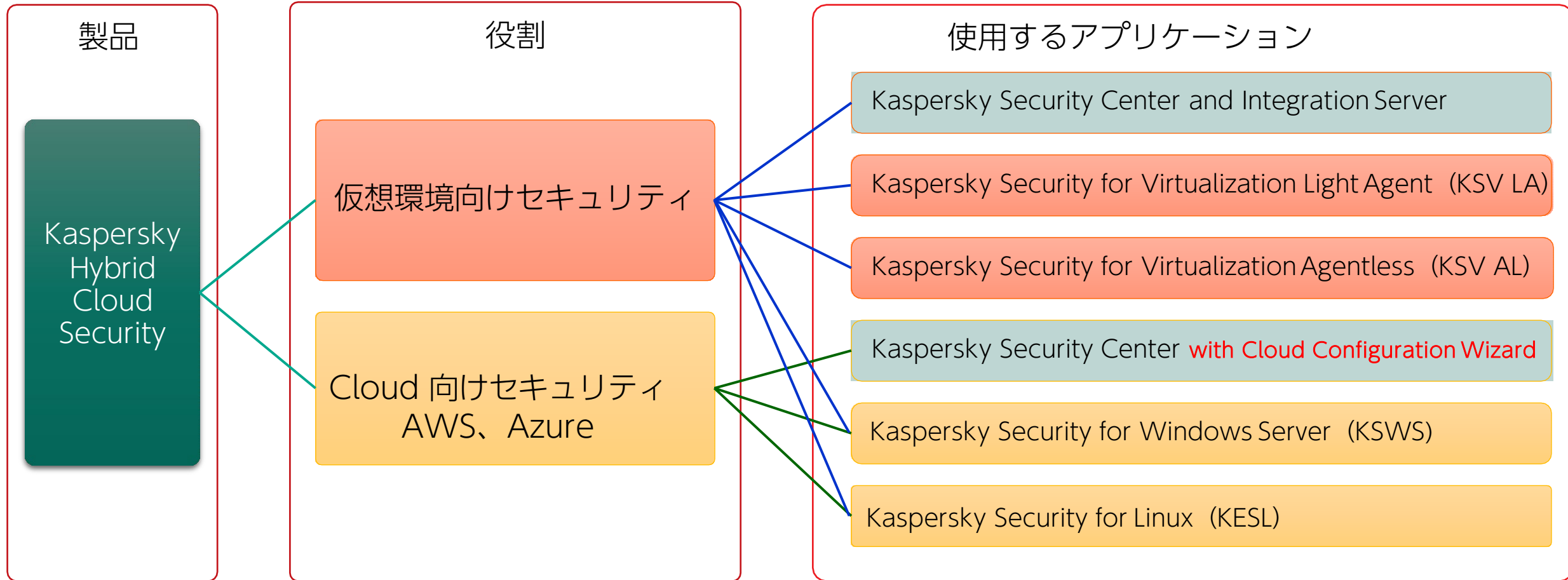
ひとつのライセンスで
マルチアプリケーション

Kaspersky Security for Virtualization

Kaspersky Security for Windows Server

Kaspersky Endpoint Security for Linux

Kaspersky Hybrid Cloud Security (KHCS)



Kaspersky Endpoint Security for Windowsは使用出来ません。

仮想環境におけるクライアントOS保護には、Kaspersky Security for Virtualizationを使用します。

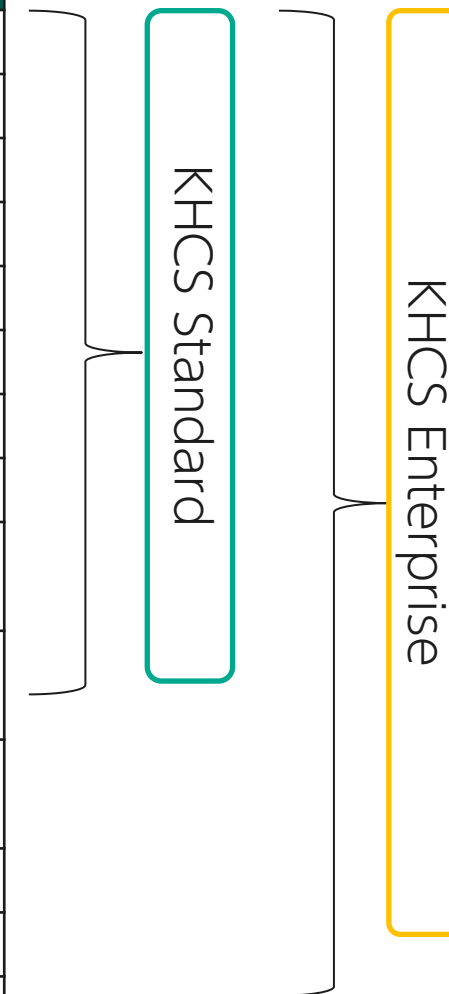
Kaspersky Hybrid Cloud Security ライセンスの種類

Kaspersky Hybrid Cloud Securityには5つのライセンス種別があります。

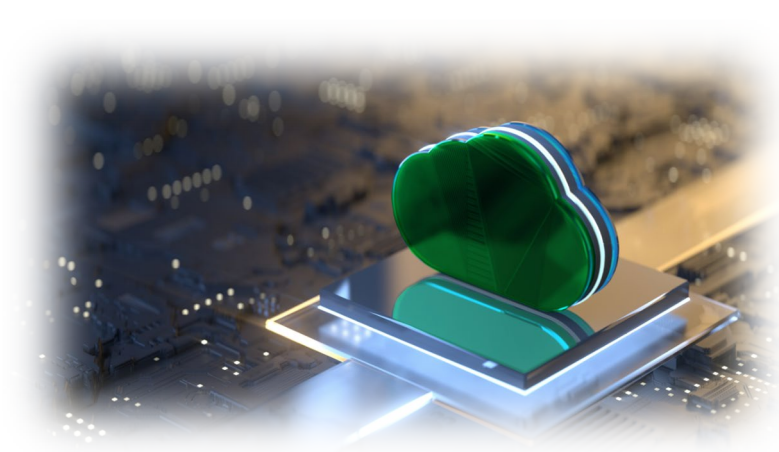
- Kaspersky Hybrid Cloud Security - デスクトップ
 - ✓ VDI向けライセンス。VDI数でカウント。
- Kaspersky Hybrid Cloud Security Enterprise - サーバー
- Kaspersky Hybrid Cloud Security - サーバー
 - ✓ 物理サーバー、仮想サーバー インスタンス、Public Cloud インスタンスに使用。
 - ✓ サーバーインスタンス数でカウント。
- Kaspersky Hybrid Cloud Security Enterprise - CPU
- Kaspersky Hybrid Cloud Security - CPU
 - ✓ 仮想環境向けライセンス。サーバー、クライアントOS共に保護対象。
 - ✓ ハイパーバイザー CPU数でカウント。

KHCS ライセンスエディション別 アプリケーション機能表

KSC	KSV LA	KSV AL	KESL	KSWS	機能
+					Cloud API integration
	+	+	+	+	ファイルアンチウイルス
	+			+	脆弱性攻撃ブロック
	+		+	+	Firewall
	+		+	+	アンチクリプター (共有フォルダー対象)
	+		+	+	ネットワーク攻撃ブロック
	+		+	+	デバイス制御
	+		+	+	Web脅威保護
	+			+	トラフィックセキュリティーメール脅威保護 (Outlook plugin)
	+			+	トラフィックセキュリティー Webコンテンツフィルタ
	+			+	アプリケーション制御 (Default Deny) for Server
	+		+	+	ファイルインテグリティ
				+	Log Inspection
		+			IDS/IPS for VMware NSX



KHCS クラウド環境での使用



クラウド環境向け特徴

- クラウドAPIを使用した自動化



Kaspersky Security Center は、
AWS APIを使って Amazon EC2 インスタンスを操作
Azure API を使って Azure 仮想マシンを操作

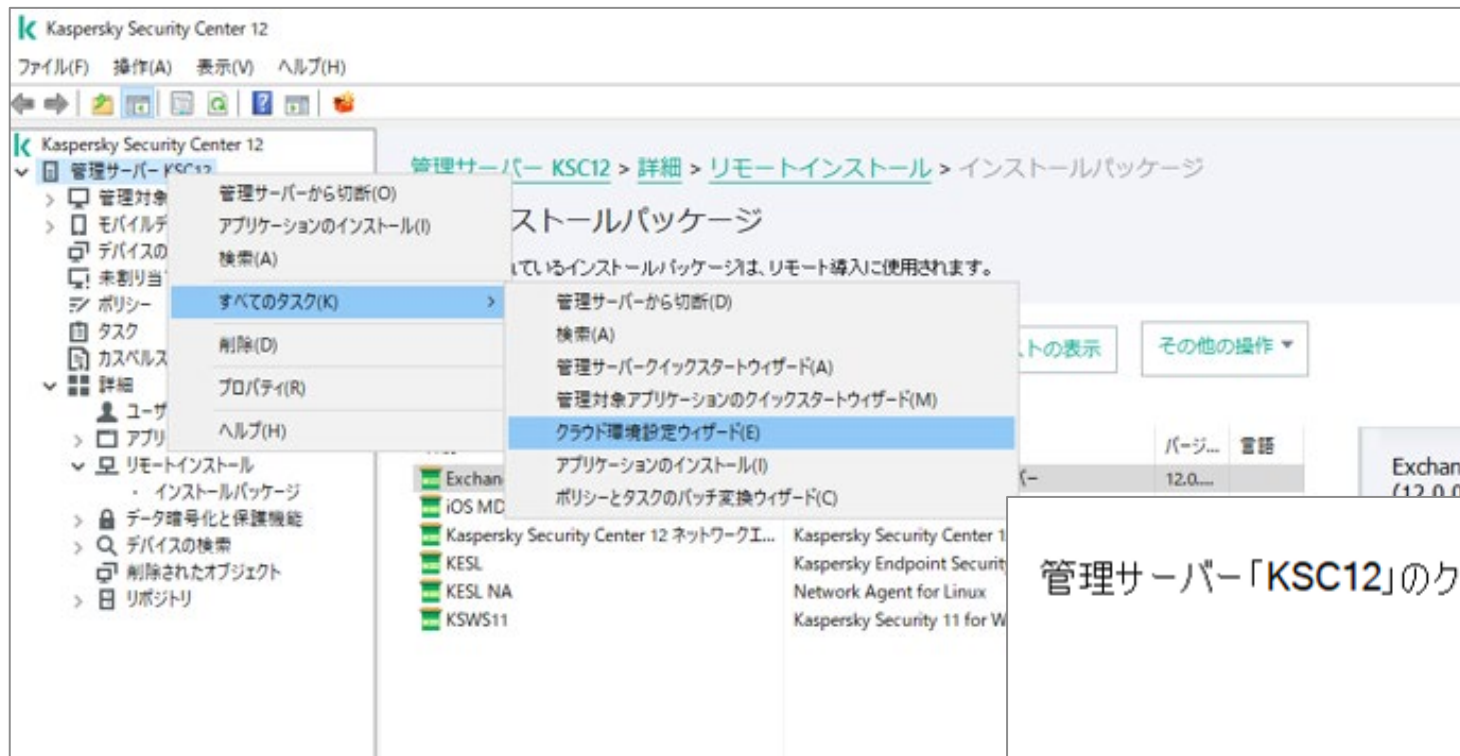
Kaspersky Security Center を
Amazon EC2 インスタンスまたは Microsoft Azure 仮想マシンに導入して、
クラウド環境内のデバイスの保護を管理可能

デプロイの自動化により、Auto Scalingに対応

- 一台から導入可能。ファイルインテグリティなど高度セキュリティも安価に導入可能。
- BYOL (Bring Your Own License : ライセンス持ち込み)
社内サーバー、仮想基盤などと合わせたライセンス管理が可能。

Kaspersky Security Center ; カスペルスキー製品を統合管理する管理サーバーアプリケーション

Cloud Configuration ウィザード



管理サーバー「KSC12」のクラウド環境設定ウィザードへようこそ

このウィザードは、管理サーバーをクラウド環境で動作するように初期設定します。Kaspersky Security Center 12 と Amazon Web Services との接続を設定するには、AWS IAM ロールか AWS IAM アクセスキーが必要です。Kaspersky Security Center 12 と Microsoft Azure との接続を設定するには、Azure サブスクリプション、Azure アプリケーション ID、Azure アプリケーションパスワードが必要です。Kaspersky Security Center 12 と Google Cloud との接続を設定するには、Google プロジェクト ID、クライアントメールアドレス、秘密鍵が必要です。

Cloud Configuration ウィザードとは

このウィザードを使用して Kaspersky Security Center を設定する場合に必要な項目

AWS クラウド環境で使用する場合

- クラウドセグメントをポーリングする権限が付与された IAM ロール
- またはクラウドセグメントをポーリングする権限が付与された IAM ユーザーアカウント

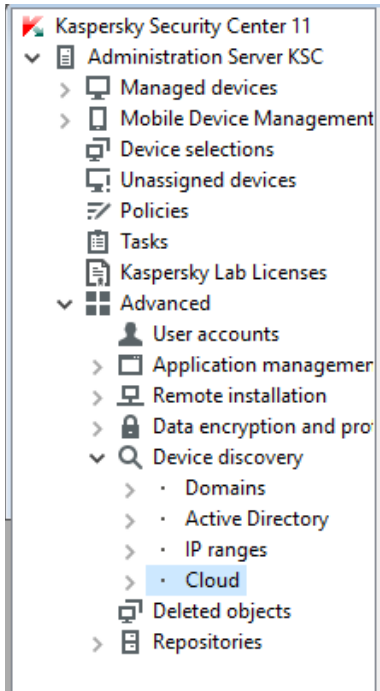
Microsoft Azure クラウド環境で使用する場合

- Azure アプリケーション ID パスワードとサブスクリプション

ウィザードでは、次のオブジェクトを作成

- 既定の設定が指定されたネットワークエージェントポリシー
- Kaspersky Endpoint Security for Linux のポリシー
- Kaspersky Security for Windows Server のポリシー
- インスタンス用の管理グループとインスタンスを自動的に管理グループに移動するためのルール
- 管理サーバーデータのデータバックアップタスク
- Linux と Windows を実行しているデバイスに保護をインストールするためのタスク
- 各管理対象デバイスに対するタスク：
 - 簡易ウイルススキャン
 - アップデートのダウンロード

クラウドポーリング



Administration Server KSC > Advanced > Device discovery > Cloud

Cloud

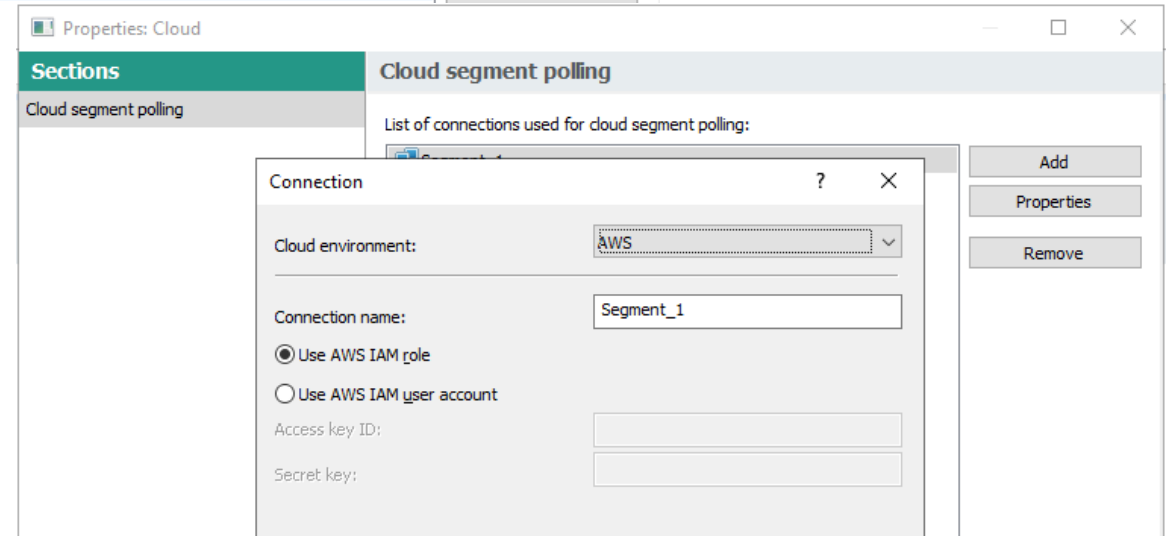
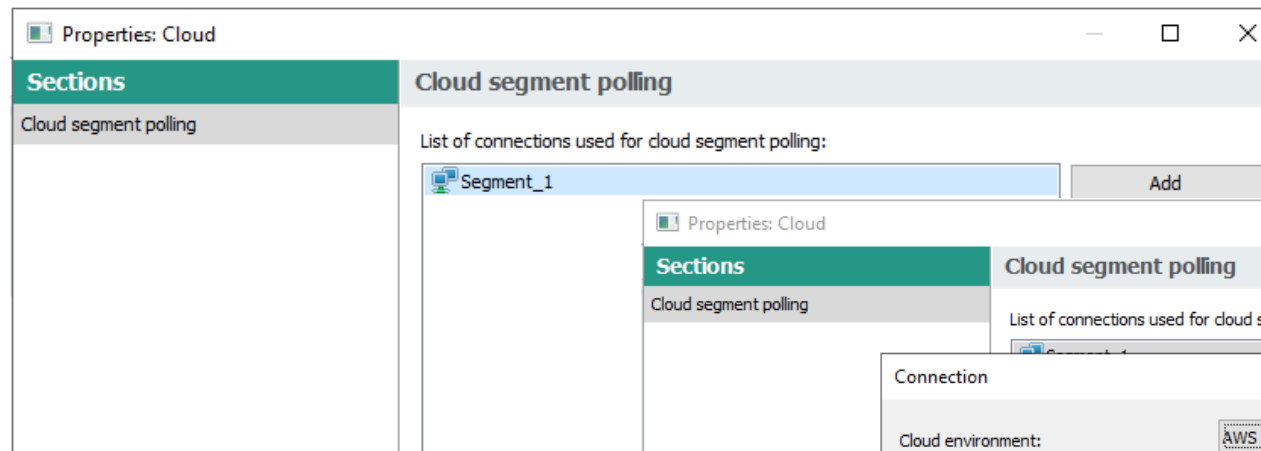
Displays the results of cloud segment poll

Actions

- Refresh
- Configure polling

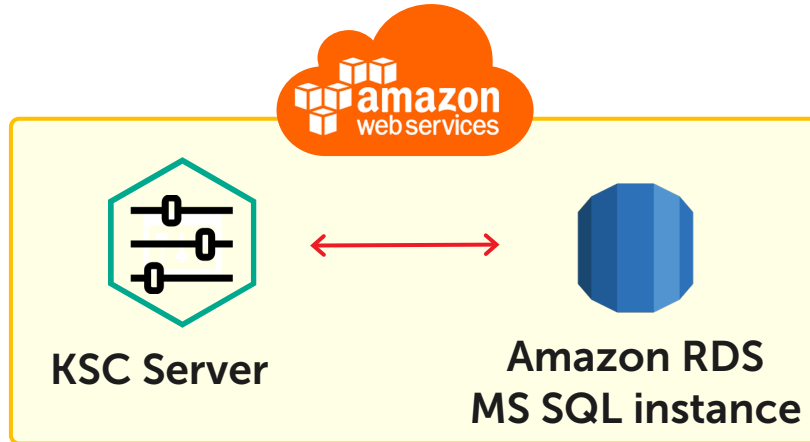
管理サーバーがクラウドセグメントのデバイスに関する情報を受信できるように、クラウドセグメントをポーリング

新しいインスタンスまたは仮想マシンが自動的に検出

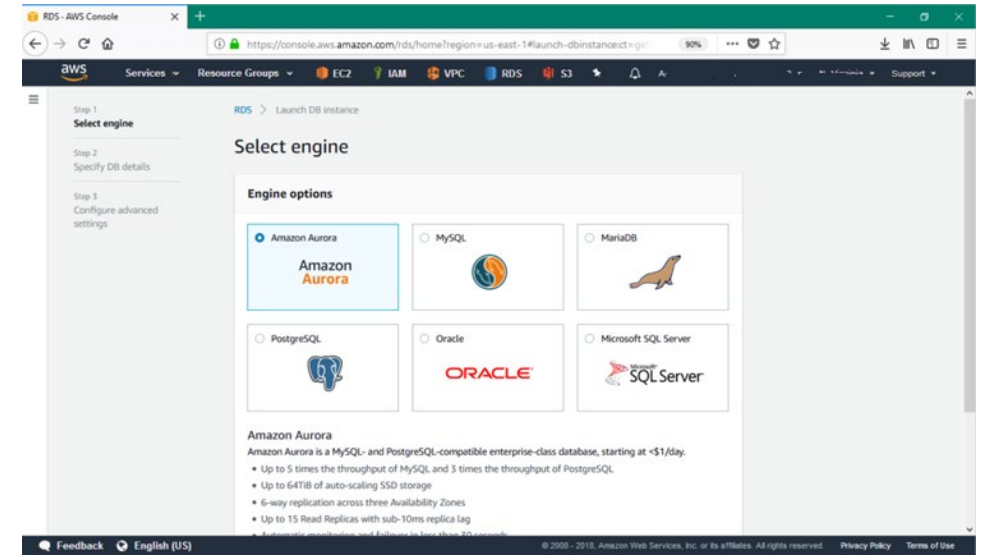


クラウドネイティブデータベースとの統合

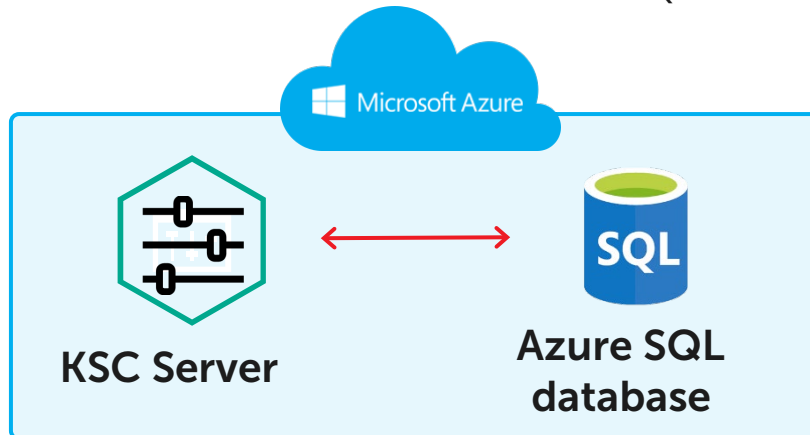
KSC Server + Amazon DRS



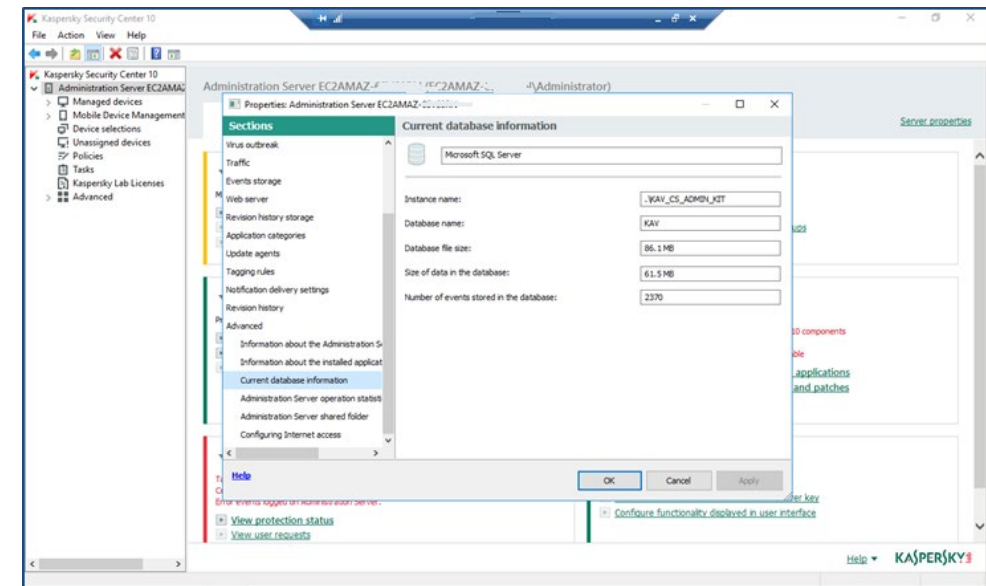
AWS Relational Database Service



KSC Server + Azure SQL



KSC Database configuration



Cloudインスタンスに対するアプリケーション自動インストール

Kaspersky Security Center 12 では次のシナリオをサポート

- クライアントデバイスが API によって検出され、製品のインストールも API によって実行される。
AWS と Azure のクラウド環境では、このシナリオがサポートされる。
- クライアントデバイスが Google API によって検出され、製品のインストールが Kaspersky Security Center によって実行される。
Google Cloud では、このシナリオのみがサポートされる。
- クライアントデバイスが Active Directory のポーリング、Windows ドメインのポーリング、IP アドレス範囲のポーリングのいずれかで検出され、製品のインストールが Kaspersky Security Center によって実行される。

Cloudインスタンスに対するアプリケーション自動インストール

AWS クラウド環境

IAM ロールとEC2 インスタンスにインストールされている Systems Manager Agent により、Kaspersky Security Center は 管理者に毎回確認しなくても、デバイスおよびデバイスのグループに自動的にアプリケーションをインストールできる。

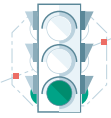
Microsoft Azure クラウド環境

Azure 仮想マシンエージェントにより、自動インストール。

Azure アプリケーション ID に次のロールが付与されている。

- Reader (ポーリングを使用して仮想マシンを検出するために必要)
- Virtual Machine Contributor (仮想マシンに保護を導入するために必要)
- SQL Server Contributor (Microsoft Azure 環境で SQL データベースを使用するために必要)

Cloudインスタンスに対するアプリケーション自動インストール



KSCによる保護コンポーネントインストール

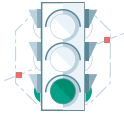
5分ごとのAWS polling

The screenshot shows the 'Properties: Cloud' window with the 'Cloud segment polling' section selected. A 'Connection' dialog box is open, showing 'AWS' as the cloud environment and 'Dev' as the connection name. A 'Schedule' dialog box is also open, showing a scheduled start of 'Every N minutes' with a start interval of 5 minutes and starting from 9:24:11 AM. The 'Run missed tasks' checkbox is checked.

「常にインストール」

The screenshot shows the 'Properties: Install protection for Windows' window with the 'Settings' section selected. The 'Installation packages' section shows the main and network agent packages. The 'Force installation package download' section has 'Using Network Agent' checked. The 'Behavior for devices managed through other Administration Servers' section has 'Install always' selected. The 'Do not re-install application if it is already installed' and 'Verify operating system version before uploading' checkboxes are also checked.

Cloudインスタンスに対するアプリケーション自動インストール



AWS CloudFormation等を使用したスクリプトベース

```
winsrv_asg.ps1 X
1  $ServerAddress = "XXXXXX";
2  $ServerPort = "14000";
3  $ServerSSLPort = "13000";
4
5  $KSWSLocalFile = "C:\Distributive\ksws.zip";
6  $KSWSRemoteFile = "http://" + $ServerAddress + ":8060/Public/KSWs.zip";
7
8  New-Item -ItemType directory -Path "C:\Distributive%";
9
10 $WC = New-Object System.Net.WebClient;
11 $FTPURI = New-Object System.Uri($KSWSRemoteFile);
12 $WC.Credentials = New-Object System.Net.NetworkCredential($Username, $Password);
13 $WC.DownloadFile($FTPURI, $KSWSLocalFile);
14
15 $ShellApp = new-object -com shell.application;
16 $Zip = $ShellApp.Namespace($KSWSLocalFile);
17 foreach ($item in $Zip.items()) {
18     $ShellApp.Namespace("C:\Distributive%").copyhere($item);
19 }
20
21 $KnagentArgs = "/norestart", "/i", "C:\Distributive\agent\klnagent.msi", "/!%xv", "C:\Distributive\agent\klnagent.log", "/qn", "EULA=1", "PRIVACYPOLICY=1", "SER
22 Start-Process msixexec -ArgumentList $KnagentArgs -PassThru -Wait;
23 Start-Sleep -s 10;
24
25 $KNAService = Get-Service klnagent -ErrorAction Continue;
26 Start-Service -Name $KNAService.Name -ErrorAction SilentlyContinue;
27
28 $KSWArgs = "/norestart", "/i", "C:\Distributive\server\ks4ws_x64.msi", "/!%xv", "C:\Distributive\ksws.log", "/qn", "EULA=1", "PRIVACYPOLICY=1", "ADDDLOCAL
29 Start-Process msixexec -ArgumentList $KSWArgs -PassThru -Wait;
30 $ConsoleArgs = "/norestart", "/i", "C:\Distributive\server\ks4wstools_x64.msi", "/!%xv", "C:\Distributive\kswsconsole.log", "/qn", "EULA=1", "PRIVACYPOLIC
31 Start-Process msixexec -ArgumentList $ConsoleArgs -PassThru -Wait
32
```



AWS Cloud Formation



Jenkins

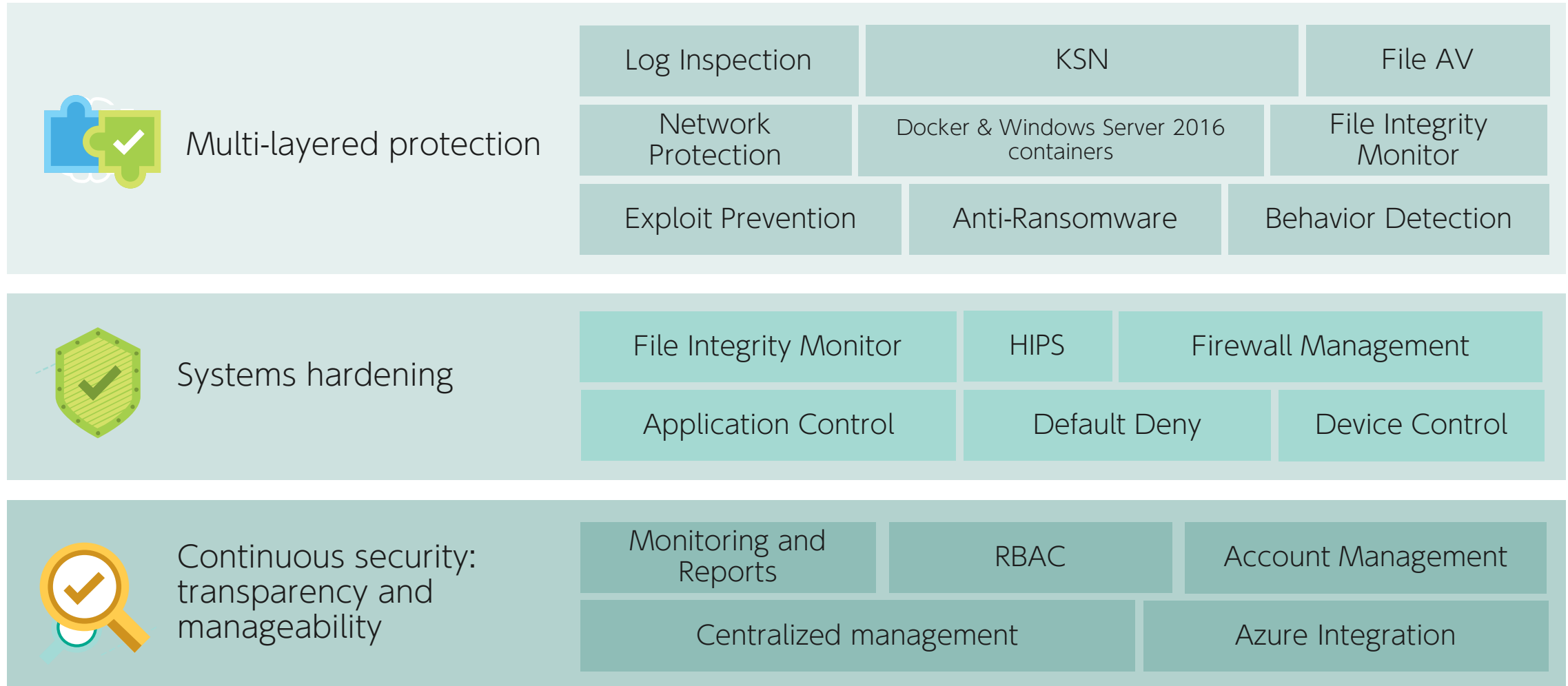


Octopus Deploy

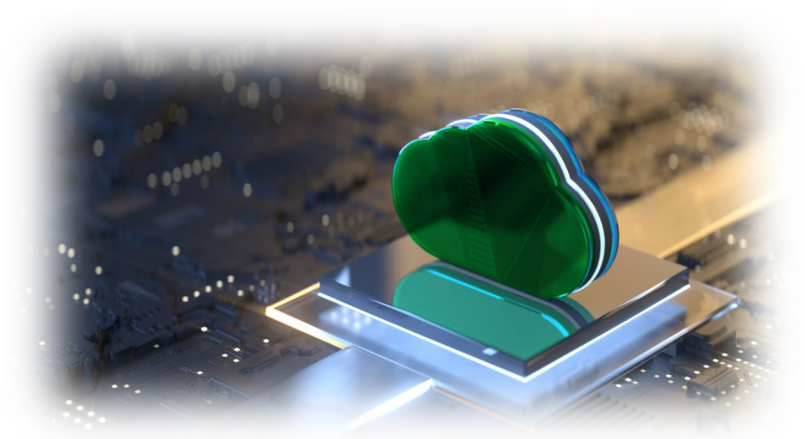
保護を展開するための展開スクリプト

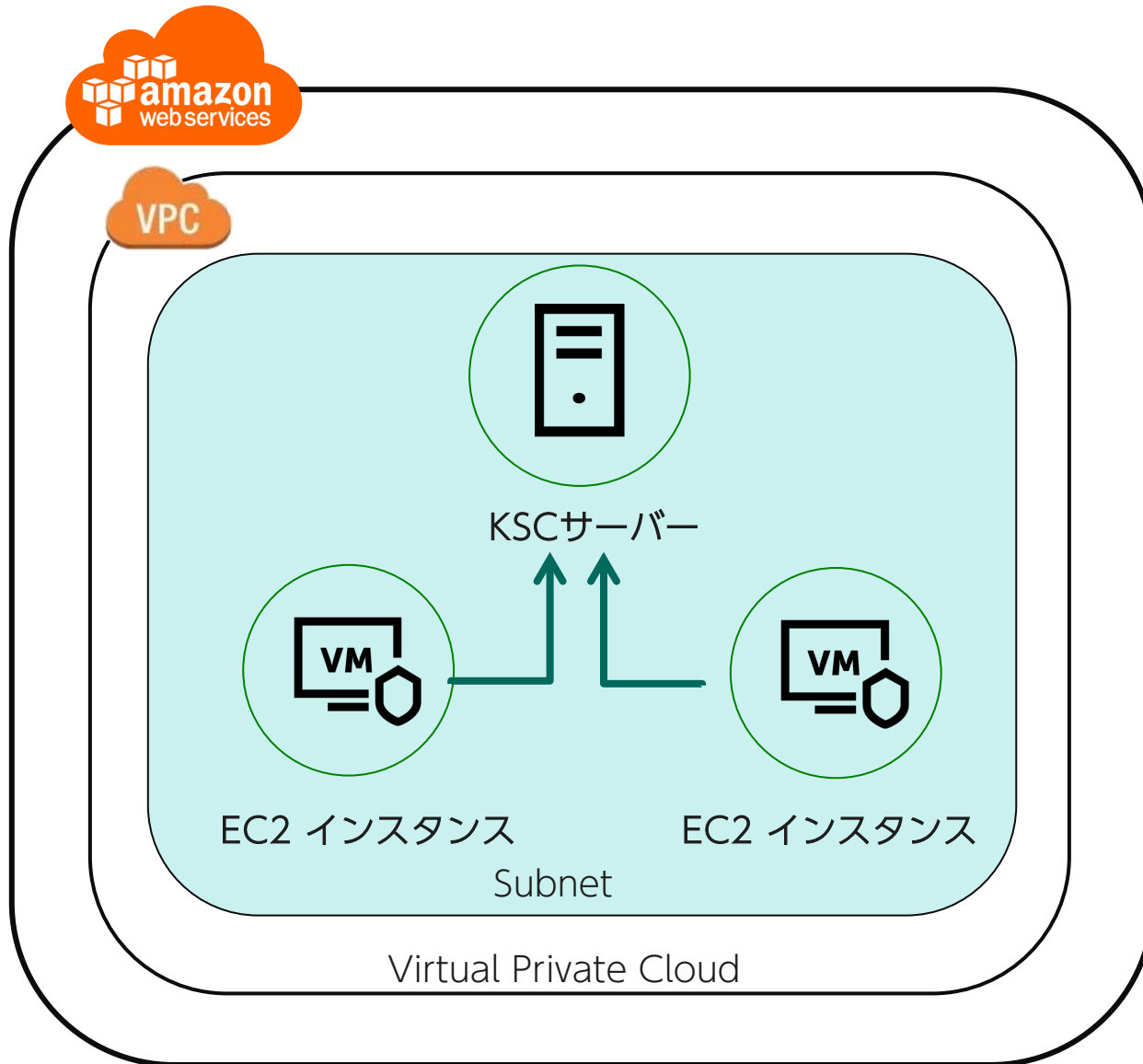
<https://support.kaspersky.co.jp/14713>

Kaspersky CWPP (クラウドワークロード保護プラットフォーム) コンポーネント

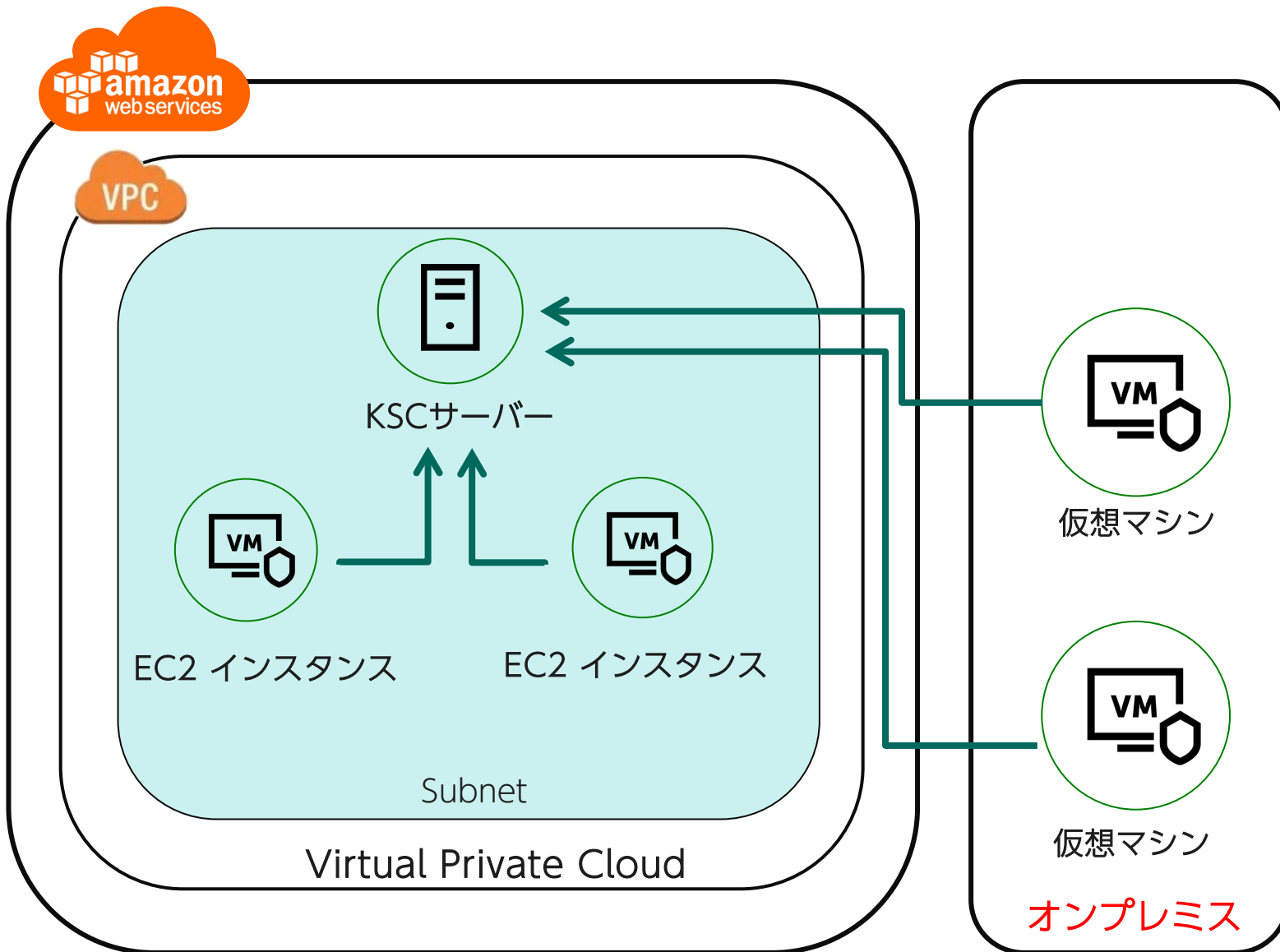


KHCS クラウド環境 構成例





- 内部トラフィックに課金発生せず
- 外側からのKSCポートへのアクセス権限が不要
- IAMロールを通じたAPI使用の自然な構成
- 仮想マシンごとのライセンス
- お客様使用のライセンスが利用可能
(ストアで購入する必要がない)



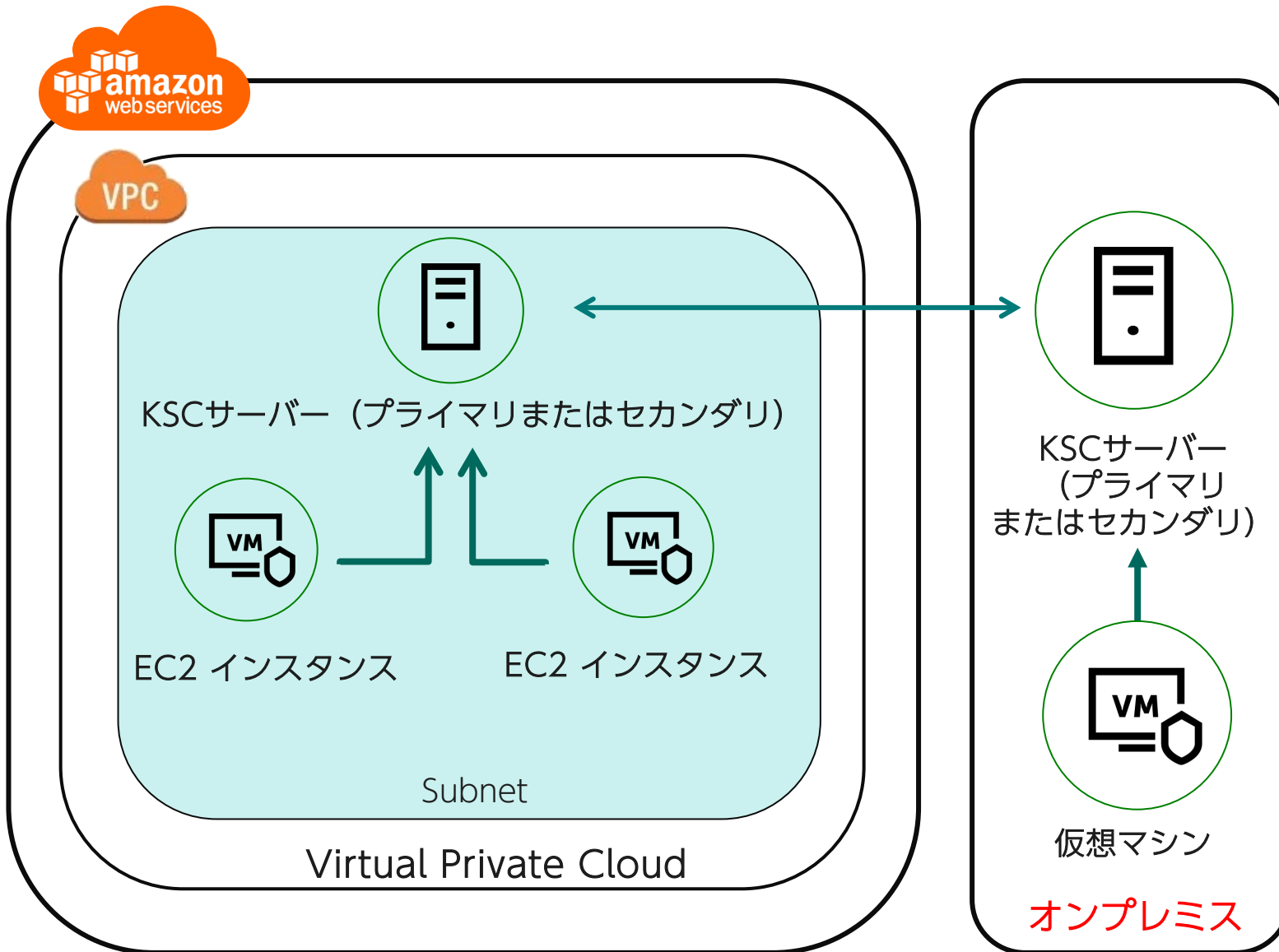
KSCアウトバウンドトラフィック
への課金

外側からのKSCポートへのアクセス権限

API使用の自然な構成

仮想マシンごとのライセンス

お客様使用のライセンスが利用可能
(ストアで購入する必要がない)



API使用の自然な構成

クラウドからの最小限アウトバウンド
トラフィック

お客様使用のライセンスが利用可能
(ストアで購入する必要がない)

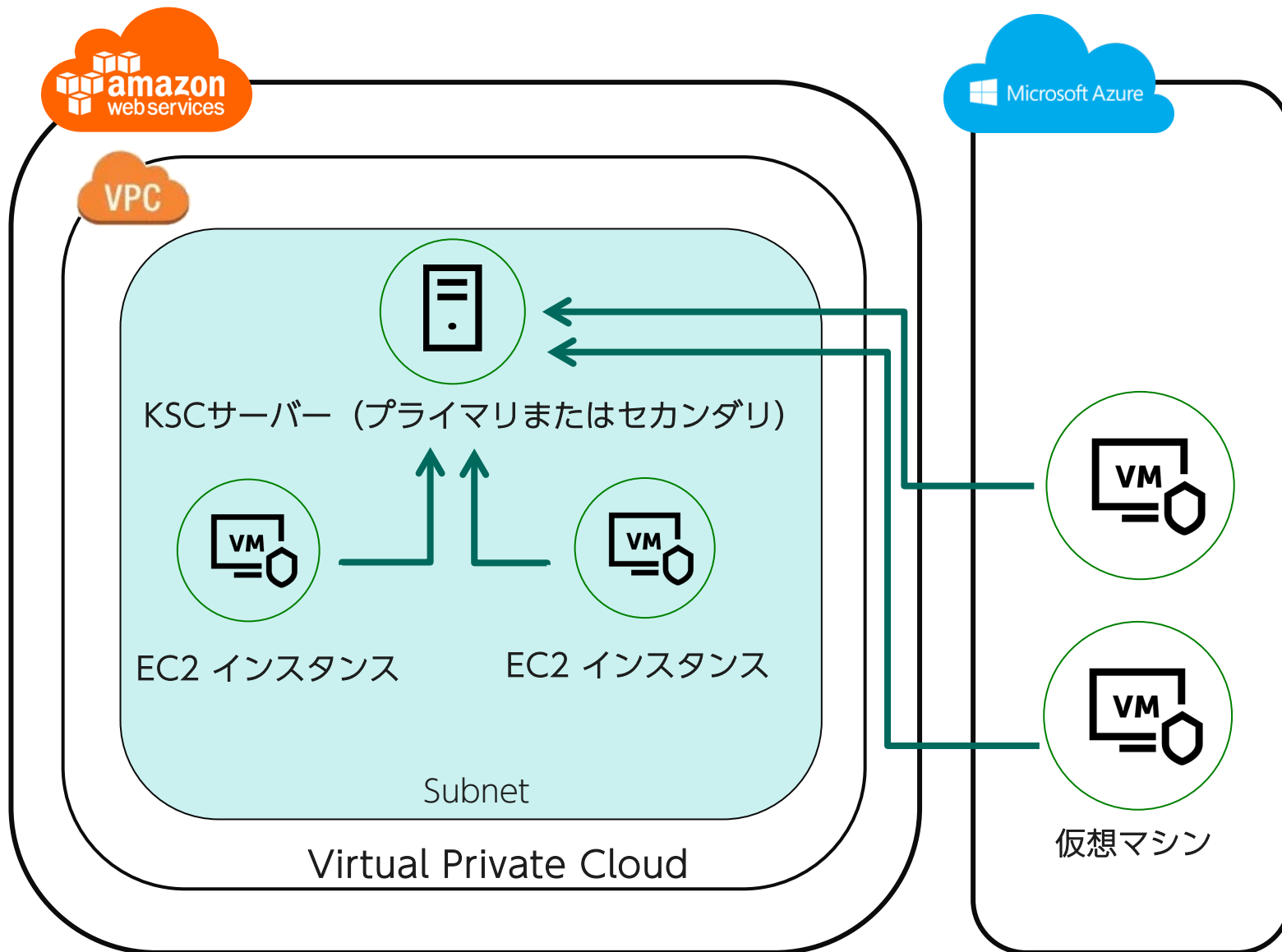
デメリット

イベントデータベースの分断

管理コンソールの分断

仮想マシン

オンプレミス



KSCアウトバウンドトラフィック
への課金

他クラウドとの統合に必要な設定

仮想マシンごとのライセンス

お客様使用のライセンスが利用可能
(ストアで購入する必要がない)