Workgroup: Internet Engineering Task Force
Internet-Draft:
Confidential Virtual Machine Provisioning in
Cloud Environment
Published: 1 December 2023
Intended Status: Informational
Expires: 3 June 2024
Authors: J. Deng  G. Yu

# Confidential Virtual Machine Provisioning in Cloud Environment

## Abstract

This document specifies the procedures of provisioning confidential virtual machine in the cloud environment.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 3 June 2024.

## Copyright Notice

**Table of Contents**



## 1. Introduction

Confidential computing protects workload and data in use leveraging hardware-based security technology. Confidential virtual machine (CVM) in the cloud environment is one use case of confidential computing. There is an increasing adoption of CVMs in the cloud. CVM allows a cloud tenant to protect the sensitive workload and data, and manage the cryptography keys independently from the cloud service providers.

When adopting CVMs in the cloud, the CVM features, CVM provisioning and management of cryptography keys, etc. depend on different hardware. Common CVM provisioning procedures and requirements are needed. This document specifies the procedures of provisioning CVMs in the cloud environment and the requirements.

## 2. Terms

The following terms are used in this document.

*CVM Platform: CVM Platform is provided and maintained by cloud service provider. It provides interfaces for cloud tenant to create and manage CVM instances on the cloud. It receives the CVM related requests from cloud tenant and interacts with cloud resource manager to perform CVM creation, decommission, update, migration etc. on the cloud infrastructure.

*Key Agent: Key Agent is the component within a CVM instance that interacts with Key Server to allocate, acquire and update security keys, and also provides and update Security Version Number (SVN) of the CVM.

*Security Version Number (SVN): SVN represents the security features of the hardware of CVM.

*Key Server: Key Server authenticates Key Agent, responds to Key Agent's requests to generate, update and return security keys, and update CVM's SVN.

*Security Key(SK): SK is allocated by Key Server under the request of Key Agent. SK is used by CVM instance to encrypt sensitive data.

*KeyID: KeyID identifies a security key.

## 3. Procedures of CVM Provisioning in Cloud Environment

The procedures of CVM provisioning in Cloud Environment includes the following:

*Feature Acquirement: Cloud Tenant acquires the CVM related features that are provided by CVM Platform.

*CVM Creation: Cloud tenant requests CVM instance(s) with selected features and CVM platform creates the requested CVM instance(s), and returns the results to the cloud tenant.

*Key Provisioning: the Key Agent in CVM instance obtains and updates security keys bounded to the CVM through communication with the Key Server.

*CVM management: Cloud tenant performs various management tasks on CVM instances, such as CVM updates, live migration, CVM decommission etc., through interacting with CVM Platform

### 3.1. Feature Acquirement

Before creating a CVM instance, Cloud Tenant acquires the supported CVM features from CVM platform. Figure 1 shows example feature acquirement between Cloud Tenant and CVM Platform. CVMFeatureRequest message is sent by Cloud Tenant to CVM Platform requesting the supported CVM features. CVM returns CVMFeatureResponse carrying a list of supported features, which may include:

*SecureBoot: whether secure boot is supported.

*LiveMigration: whether live migration is supported.

*AuxilaryFirmware: whether allows Cloud Tenant to specify firmware to be used.

*BIOS: whether allows Cloud Tenant to customize BIOS.

*SVN: whether allows Cloud Tenant to specifies SVN.

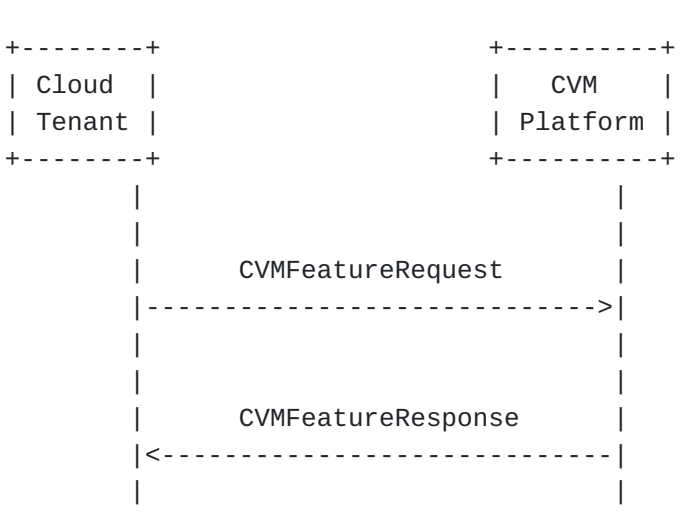


Figure 1: CVM feature acquirement

### 3.2. CVM Creation

Figure 2 shows that Cloud Tenant requests to create CVM instance(s) and CVM Platform responds with the creation result. In the CVMCreateRequest message requesting CVM creation, Client tenant provides the requested features. The features are described as in Section 3.1. CVM Platform returns with CVMCreateResponse. If the creation is successful, in CVMCreateResponse message, CVM Platform indicated successful CVM creation and returns information on the features requested by Cloud Tenant.

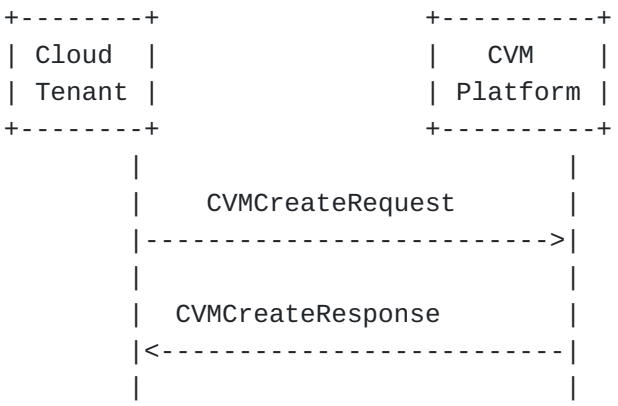


Figure 2: CVM instance creation

### 3.3. Key Provisioning

The key provisioning consists of Policy Setup, Key Allocation, Key Acquirement, and Key Update between Key Agent and Key Server.

*Policy Setup: Cloud Tenant provide Key Server with Keying Policy.

*Key Allocation: Key Agent requests Key Server to allocate a new security key.

*Key Acquirement: Key Agent obtains a pre-allocated security key by providing the KeyID to Key Server.

*Key Update: Key Agent updates its SVN with Key Server.

The security considerations for the communication between Key Agent and Key Server are presented in Section 5.

#### 3.3.1. Policy Setup

In Policy Setup, Cloud Tenant provides Key server with information needed for Key Allocation, Key Acquirement, and Key Update. The information at least includes SVN, measurements, etc. Figure 3 shows example Keying Policy setup between Key Agent and Key Server.


Figure 3: Key provisioning

### 3.3.2. Key Allocation

Figure 4 shows example key allocation. Key Agent sends KeyAllocRequest message to Key Server to request a new security key. Key Server then allocates a KeyID, generates and saves a root key for Key Agent, derives a security key from the root key with input parameters including the SVN provided by Key Agent, and returns the KeyID and Security Key. Allocation usually occurs when CVM is started for the first time, and CVM needs to use Security Key for encryption.


Figure 4: Key provisioning

### 3.3.3. Key Acquirement

Figure 5 shows example key acquirement where Key Agent acquires a pre-allocated Security Key with the KeyID.

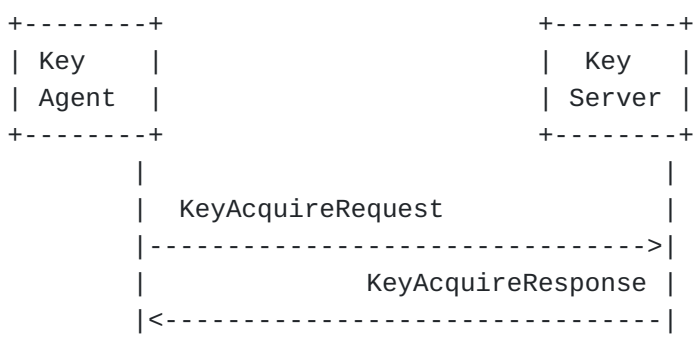


Figure 5: Key provisioning

### 3.3.4. Key Update

Key Agent within a CVM may chose to update the minimal required SVN of the Key by sending KeyUpdateRequest to Key Server. Key Server will only update the SVN if the old SVN with the Key Agent is lower than the target SVN. After successful SVN update, a Key Agent with outdated SVN cannot acquire the Security Key with the pre-allocated KeyID. A CVM which meets the requirement of minimum SVN can request the Key Server to re-allocate a new Security Key from the corresponding root key. Figure 6 shows example key update.

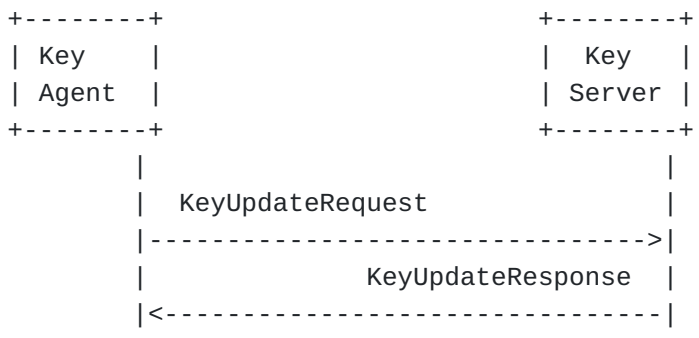


Figure 6: Key Update

## 3.4. CVM Management

This section presents the procedures for CVM management.

# 4. IANA Considerations

This memo includes no request to IANA.

## 5. Security Considerations

### 5.1. Communication Security Between Key Agent and Key Server

Key Agent and Key server are mutually authenticated and the communications between them are confidentially and integrity protected. The security can leverage the attestation evidence in [RFC9334]. The messages can use CBOR and the security wrapper as in [RFC9052].

### 5.2. Communication Security Between Cloud Tenant and Key Server

This section considers the communication security between Cloud Tenant and Key Server.

## 6. References

### 6.1. Normative References


**[RFC9334]** Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <https://www.rfc-editor.org/info/rfc9334>.

**[RFC9052]** Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", STD 96, RFC 9052, DOI 10.17487/RFC9052, August 2022, <https://www.rfc-editor.org/info/rfc9052>.


## Acknowledgements

## Contributors

## Authors' Addresses


Juan Deng

Email: dengjuan.deng@alibaba-inc.com

Guorui Yu

Email: ruogui.ygr@alibaba-inc.com