+++ title = “AMS KMS” description = “Using AWS KMS to encrypt database secrets” keywords = [“grafana”, “AWS KMS integration”] weight = 3 +++

Using AWS KMS to encrypt database secrets

You can use an encryption key from AWS Key Management Service to encrypt secrets in the Grafana database.

Prerequisites:

- An AWS account with permission to view and create KMS keys and programmatic credentials to access those keys
- Access to the Grafana [configuration]({{< relref “../../administration/configuration/#config-file-locations” >}}) file

1. Create a symmetric API key either from the AWS Management Console or by using the AWS KMS API. For detailed instructions, refer to Creating keys.
2. Retrieve the Key ID. In AWS terms, this can be a key ID, a key ARN (Amazon Resource Name), an alias name, or an alias ARN. For more information about how to retrieve a key ID from AWS, refer to Finding the key ID and key ARN.
3. Create a programmatic credential (access key ID and secret access key), which has permission to view the key that you created. In AWS, you can control access to your KMS keys by using key policies, IAM policies, and grants. You can also create temporary credentials, which must provide a session token along with an access key ID and a secret access key.
4. From within Grafana, turn on [envelope encryption]({{< relref “../../administration//database-encryption.md” >}}).
5. Add your AWS KMS details to the Grafana configuration file; depending on your operating system, it is usually named `grafana.ini`: a. Add a new section to the configuration file, with a name in the format of `[security.encryption.awskms.<KEY-NAME>]`, where `<KEY-NAME>` is any name that uniquely identifies this key among other provider keys. b. Fill in the section with the following values:

    - `key_id`: a reference to a key stored in the KMS. This can be a key ID, a key Amazon Resource Name (ARN), an alias name, or an alias ARN. If you are using an alias, use the prefix `alias/`. To specify a KMS key in a different AWS account, use its ARN or alias. For more information about how to retrieve a key ID from AWS, refer to Finding the key ID and key ARN. | `key_id` option | Example value | | — | — | | Key ID | `1234abcd-12ab-34cd-56ef-1234567890ab` | | Key ARN | `arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`

| | Alias name | `alias/ExampleAlias` | | Alias ARN | `arn:aws:kms:us-east-2:111122223333:alias` |

- `access_key_id`: The AWS Access Key ID that you previously generated.
- `secret_access_key`: The AWS Secret Access Key you previously generated.
- `region`: The AWS region where you created the KMS key. The region is contained in the key's ARN. For example: `arn:aws:kms:*us-east-2*:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`

An example of an AWS KMS provider section in the `grafana.ini` file is as follows:

```
# AWS key management service provider setup
;[security.encryption.awskms.example-encryption-key]
# Reference to a KMS key - either key ID, key ARN, alias name, or ARN
;key_id = 1234abcd-12ab-34cd-56ef-1234567890ab
# AWS access key ID
;access_key_id = AKIAIOSFODNN7EXAMPLE
# AWS secret access key
;secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
# AWS region, for example eu-north-1
;region = eu-north-1
```

6. Update the `[security]` section of the `grafana.ini` configuration file with the new Encryption Provider key that you created:

   ```
   [security]
   # previous encryption key, used for legacy alerts, decrypting existing secrets or used
   secret_key = AaaaAaaa
   # encryption provider key in the format <PROVIDER>.<KEY_NAME>
   encryption_provider = awskms.example-encryption-key
   # list of configured key providers, space separated
   available_encryption_providers = awskms.example-encryption-key
   ```

   > **Note:** The encryption key that is stored in the `secret_key` field is still used by Grafana's legacy alerting system to encrypt secrets, for decrypting existing secrets, or it is used as the default provider when external providers are not configured. Do not change or remove that value when adding a new KMS provider.

7. Restart Grafana.

8. (Optional) From the command line and the root directory of Grafana, re-encrypt all of the secrets within the Grafana database with the new key using the following command:

   ```
   grafana-cli admin secrets-migration re-encrypt
   ```

If you do not re-encrypt existing secrets, then they will remain encrypted by the previous encryption key. Users will still be able to access them.

> **Note:** This process could take a few minutes to complete, depending on the number of secrets (such as data sources or alert notification channels) in your database. Users might experience errors while this process is running, and alert notifications might not be sent.

> **Note:** If you are updating this encryption key during the initial setup of Grafana before any data sources, alert notification channels, or dashboards have been created, then this step is not necessary because there are no secrets in Grafana to migrate.