bitwarden | Hilfezentrum Artikel

SECRETS MANAGER > INTEGRATIONEN

Ansible

Ansicht im Hilfezentrum:
https://bitwarden.com/help/ansible-integration/

Ansible

Bitwarden bietet eine Integration mit Ansible an, um Geheimnisse aus dem Secrets Manager abzurufen und in Ihr Ansible-Playbook einzufügen. Das Lookup-Plugin wird abgerufene Geheimnisse als maskierte Umgebungsvariablen in ein Ansible-Playbook einfügen. Um die Sammlung einzurichten:

Anforderungen

- Wir empfehlen die Installation von Python-Paketen in einer Python virtuellen Umgebung.
- Aktuelle Version von Ansible, die auf Ihrem System installiert ist.
- Bitwarden Secrets Manager mit einem aktiven Service-Konto.

Bevor Sie die Ansible-Sammlung einrichten, empfehlen wir Ihnen auch, den Secrets Manager zu öffnen, um auf Ihren Zugriffstoken und alle Geheimnisse zuzugreifen, die Sie in die Einrichtung einbeziehen möchten.

Installieren Sie die Bitwarden Ansible Sammlung

Die folgende Anleitung ist ein Einrichtungsbeispiel für die Bitwarden-Sammlung mit einer Linux-Maschine.

1. Installieren Sie das Bitwarden SDK:

```Bash
pip install bitwarden-sdk
```

2. Installieren Sie die bitwarden.secrets Sammlung:

```Bash
ansible-galaxy collection install bitwarden.secrets
```

Jetzt, da die Ansible-Sammlung installiert wurde, können wir beginnen, Bitwarden-Geheimnisse von einem Ansible-Playbook mit `bitwarden.secrets.lookup` aufzurufen. Der folgende Abschnitt wird Beispiele enthalten, um diesen Prozess zu demonstrieren.

> **Note**
>
> macOS-Benutzer müssen möglicherweise die folgende Umgebungsvariable in der Shell setzen, um Ansible-Probleme flussaufwärts zu vermeiden.
>
> - `Export OBJC_DISABLE_INITIALIZE_FORK_SAFETY=JA`

Hole Bitwarden Geheimnisse

Um Geheimnisse aus dem Secrets Manager in Ihrem Playbook abzurufen, gibt es zwei Methoden:

Speichern Sie das Zugriffs-Token als Umgebungsvariable.

Mit dem Secrets Manager können wir unser Zugriffstoken sicher als Umgebungsvariable in der Shell setzen und das Playbook verwenden, um das Geheimnis abzurufen. So authentifizieren Sie das Zugriffstoken :

1. Im Shell führen Sie den folgenden Befehl aus, um Ihre Umgebungsvariable für den Zugriffstoken zu setzen:

```Bash
export BWS_ACCESS_TOKEN=<ACCESS_TOKEN_VALUE>
```

2. Jetzt, da die Umgebungsvariable festgelegt wurde, können wir das Lookup-Plugin verwenden, um Variablen in unserem Playbook zu füllen. Zum Beispiel:

```Bash
  vars:
        database_password: "{{ lookup('bitwarden.secrets.lookup', '<SECRET_ID>') }}"
```

> **Note**
> Indem `BWS_ACCESS_TOKEN` als Umgebungsvariable festgelegt wird, kann auf das Zugriffstoken verwiesen werden, ohne den rohen Wert des Zugriffstokens in das Playbook aufzunehmen.

Geben Sie das Zugriffstoken im Playbook ein

Der Zugriffs-Token des Secrets Manager kann auch innerhalb des Playbooks selbst referenziert werden. Diese Methode erfordert nicht, dass Sie die Umgebungsvariable `BWS_ACCESS_TOKEN` in Ihrer Shell verwenden, jedoch wird der Wert des Zugriffsstokens im Playbook selbst gespeichert.

1. Zugriffstoken können im Playbook mit dem folgenden Beispiel enthalten sein:

```Bash
vars:
    password_with_a_different_access_token: "{{ lookup('bitwarden.secrets.lookup', '<SECRET_ID_V
ALUE>',
access_token='<ACCESS_TOKEN_VALUE>') }}"
```

Mit dieser Methode können mehrere Zugriffstoken in einem einzigen Playbook referenziert werden.

Abrufen des Geheimnisses von einem anderen Server

Benutzer von selbst gehostetem Bitwarden können Geheimnisse von ihrem Bitwarden-Server abrufen, indem sie die `base_url`, `api_url` und `identity_url` einbeziehen:

```
Bash

vars:
    secret_from_other_server: "{{ lookup('bitwarden.secrets.lookup', '<SECRET_ID>', base_url='http
s://bitwarden.example.com' ) }}"
    secret_advanced: >-
      {{ lookup('bitwarden.secrets.lookup', '<SECRET_ID>',
        api_url='https://bitwarden.example.com/api',
        identity_url='https://bitwarden.example.com/identity' ) }}
```

Beispiel-Playbook

Das Folgende ist ein Beispiel für eine Playbook-Datei mit mehreren Konfigurationsoptionen.

```
Bash

---
- name: Using secrets from Bitwarden

  vars:
    bws_access_token: "{{ lookup('env', 'CUSTOM_ACCESS_TOKEN_VAR') }}"
    state_file_dir: "{{ '~/.config/bitwarden-sm' | expanduser }}"
    secret_id: "9165d7a8-2c22-476e-8add-b0d50162c5cc"

    secret: "{{ lookup('bitwarden.secrets.lookup', secret_id) }}"
    secret_with_field: "{{ lookup('bitwarden.secrets.lookup', secret_id, field='note' ) }}"
    secret_with_access_token: "{{ lookup('bitwarden.secrets.lookup', secret_id, access_token=bws_ac
cess_token ) }}"
    secret_with_state_file: "{{ lookup('bitwarden.secrets.lookup', secret_id, state_file_dir=state_
file_dir ) }}"

  tasks:
    - name: Use the secret in a task
      include_tasks: tasks/add_db_user.yml # reference the secrets with "{{ secret }}", "{{ secret_
with_field }}", etc.
```

> **Note**
> Im obigen Beispiel zeigt die `CUSTOM_ACCESS_TOKEN_VAR` , dass Sie mehrere, unterschiedliche Zugriffstoken einbeziehen können. Diese müssen nicht hart codiert sein und können sicher in Ihr Playbook geliefert werden.

Variable	Zusätzliche Informationen
`bws_zugriffstoken`	Suche auf Zugriffstoken `env` Variable.
`Dateiordner_des_Status`	Ein Verzeichnis, in dem Ihr Authentifizierungszustand zwischengespeichert werden kann.
`geheime_id`	ID des Geheimnisses, das Sie suchen möchten.
`Geheimnis`	Suchen Sie einen geheimen Wert und speichern Sie ihn als Variable mit dem Namen `"geheim"`.
`Geheimnis_mit_Feld`	Suche ein Geheimnis mit zusätzlicher Feldausgabe. In diesem Beispiel gibt die Suche den `'Notiz'` Wert des Geheimnisses zurück.
`Geheimnis_mit_Zugriffstoken`	Suchen Sie ein Geheimnis mit dem im Antrag enthaltenen Zugriffstoken-Wert.
`Geheimnis_mit_Statusdatei`	Suchen Sie ein Geheimnis mit der im Antrag enthaltenen vorab konfigurierten Zustandsdatei.

Zusätzliche Anfragen und Felder

Zusätzlich zur `secret_id` können mehrere Felder in die `bitwarden.secrets.lookup` aufgenommen werden. Das folgende JSON-Objekt enthält alle Felder, die im Playbook-Lookup referenziert werden können:

```Bash
{
  "id": "be8e0ad8-d545-4017-a55a-b02f014d4158",
  "organizationId": "10e8cbfa-7bd2-4361-bd6f-b02e013f9c41",
  "projectId": "e325ea69-a3ab-4dff-836f-b02e013fe530",
  "key": "SES_KEY",
  "value": "0.982492bc-7f37-4475-9e60",
  "note": "",
  "creationDate": "2023-06-28T20:13:20.643567Z",
  "revisionDate": "2023-06-28T20:13:20.643567Z"
}
```

Um zusätzliche Felder wie "Notiz" abzurufen, kann der folgende Befehl zum Playbook hinzugefügt werden:

```Bash
  vars:
        database_password: "{{ lookup('bitwarden.secrets.lookup', '0037ed90-efbb-4d59-a798-b1030124
87a0', field='note') }}"
```