Installation

`eksctl` is available to install from official releases as described below. We recommend that you install `eksctl` from only the official GitHub releases. You may opt to use a third-party installer, but please be advised that AWS does not maintain nor support these methods of installation. Use them at your own discretion.

Prerequisite

You will need to have AWS API credentials configured. What works for AWS CLI or any other tools (kops, Terraform, etc.) should be sufficient. You can use `~/.aws/credentials` file or environment variables. For more information read AWS documentation.

You will also need AWS IAM Authenticator for Kubernetes command (either `aws-iam-authenticator` or `aws eks get-token` (available in version 1.16.156 or greater of AWS CLI) in your `PATH`.

The IAM account used for EKS cluster creation should have these minimal access levels.

AWS Service	Access Level
CloudFormation	Full Access
EC2	**Full:** Tagging **Limited:** List, Read, Write
EC2 Auto Scaling	**Limited:** List, Write
EKS	Full Access
IAM	**Limited:** List, Read, Write, Permissions Management
Systems Manager	**Limited:** List, Read

For Unix

To download the latest release, run:

```
# for ARM systems, set ARCH to: `arm64`, `armv6` or `armv7`
ARCH=amd64
```

```
PLATFORM=$(uname -s)_$ARCH

curl -sLO "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_$PLATFORM.tar.gz"

# (Optional) Verify checksum
curl -sL "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_checksums.txt" | grep $PLATFORM | sha256sum --check

tar -xzf eksctl_$PLATFORM.tar.gz -C /tmp && rm eksctl_$PLATFORM.tar.gz

sudo mv /tmp/eksctl /usr/local/bin
```

For Windows

Direct download (latest release): AMD64/x86_64 - ARMv6 - ARMv7 - ARM64

Make sure to unzip the archive to a folder in the `PATH` variable.

Optionally, verify the checksum:

1. Download the checksum file: latest
2. Use Command Prompt to manually compare `CertUtil`'s output to the checksum file downloaded.

   ```
   REM Replace amd64 with armv6, armv7 or arm64
   CertUtil -hashfile eksctl_Windows_amd64.zip SHA256
   ```

3. Using PowerShell to automate the verification using the `-eq` operator to get a `True` or `False` result:

   ~~~
   # Replace amd64 with armv6, armv7 or arm64
    (Get-FileHash -Algorithm SHA256 .\eksctl_Windows_amd64.zip).Hash -eq ((Get-Content .\eksctl_checksums.txt) -match 'eksctl_Windows_amd64.zip' -split ' ')[0]
    ```

   #### Using Git Bash:
   ```sh
   # for ARM systems, set ARCH to: `arm64`, `armv6` or `armv7`
   ARCH=amd64
   PLATFORM=windows_$ARCH

   curl -sLO "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_$PLATFORM.zip"

   # (Optional) Verify checksum
   curl -sL "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_checksums.txt" | grep $PLATFORM | sha256sum --check
   ~~~

```
unzip eksctl_$PLATFORM.zip -d $HOME/bin

rm eksctl_$PLATFORM.zip
```

The `eksctl` executable is placed in `$HOME/bin`, which is in `$PATH` from Git Bash.

Docker

For every release and RC a container image is pushed to ECR repository `public.ecr.aws/eksctl/eksctl`. Learn more about the usage on ECR Public Gallery - eksctl. For example,

```
docker run --rm -it public.ecr.aws/eksctl/eksctl version
```

Third-Party Installers (Not Recommended)

For MacOS

Homebrew

```
brew tap weaveworks/tap
brew install weaveworks/tap/eksctl
```

MacPorts

```
port install eksctl
```

For Windows

Chocolatey

```
choco install eksctl
```

Scoop

```
scoop install eksctl
```

Shell Completion

Bash

To enable bash completion, run the following, or put it in `~/.bashrc` or `~/.profile`:

```
. <(eksctl completion bash)
```

Zsh

For zsh completion, please run:

```
mkdir -p ~/.zsh/completion/
eksctl completion zsh > ~/.zsh/completion/_eksctl
```

and put the following in `~/.zshrc`:

```
fpath=($fpath ~/.zsh/completion)
```

Note if you're not running a distribution like oh-my-zsh you may first have to enable autocompletion (and put in `~/.zshrc` to make it persistent):

```
autoload -U compinit
compinit
```

Fish

The below commands can be used for fish auto completion:

```
mkdir -p ~/.config/fish/completions
eksctl completion fish > ~/.config/fish/completions/eksctl.fish
```

Powershell

The below command can be referred for setting it up. Please note that the path might be different depending on your system settings.

```
eksctl completion powershell >
C:\Users\Documents\WindowsPowerShell\Scripts\eksctl.ps1
```

Features

The features that are currently implemented are:

- Create, get, list and delete clusters
- Create, drain and delete nodegroups
- Scale a nodegroup
- Update a cluster
- Use custom AMIs
- Configure VPC Networking
- Configure access to API endpoints

- Support for GPU nodegroups
- Spot instances and mixed instances
- IAM Management and Add-on Policies
- List cluster Cloudformation stacks
- Install coredns
- Write kubeconfig file for a cluster

Was this page helpful?

👍 👎