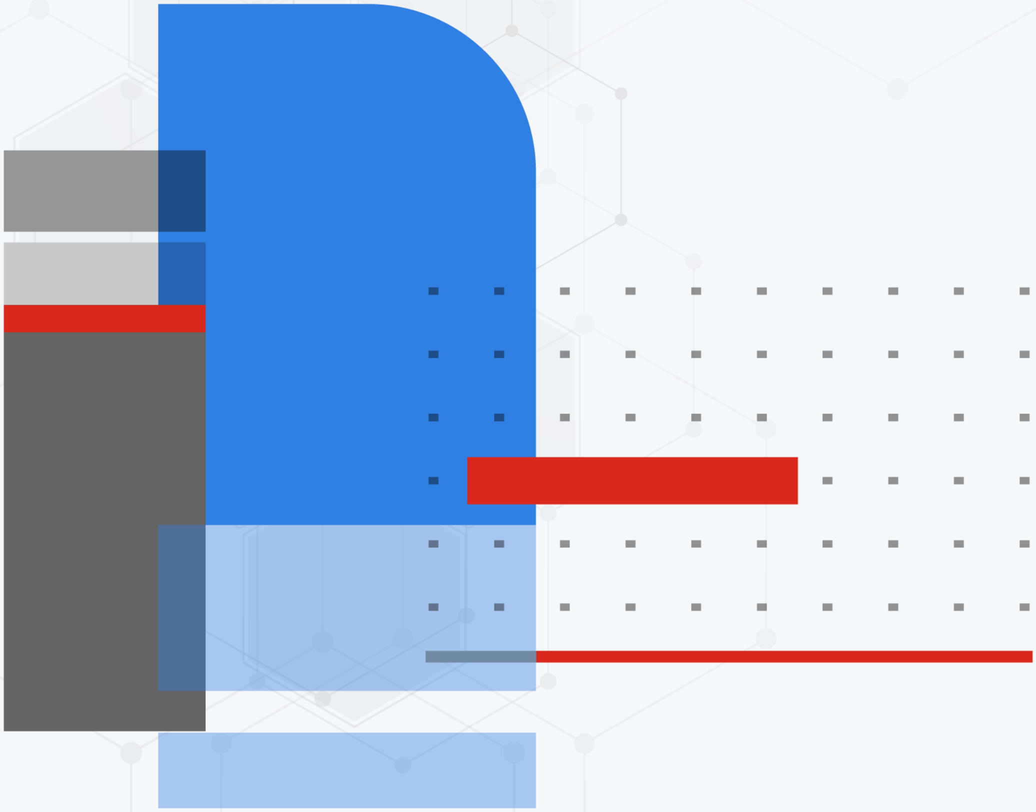
CLI Reference

FortiCNAPP 2.12.1



March 17, 2026
FortiCNAPP 2.12.1 CLI Reference
91-2121-20260317

TABLE OF CONTENTS

Change Log

Date	Change Description
2024-09-29	Initial release.
2024-10-17	1.54.0 release.
2024-10-31	2.1.1 release.
2024-11-04	2.1.2 release.
2024-11-13	2.1.3 release.
2024-12-09	2.1.4 release.
2024-12-30	Updated Get Started with the FortiCNAPP CLI.
2025-01-06	2.1.5 release.
2025-04-09	2.1.6 release.
2025-04-17	2.1.8 release.
2025-05-07	2.1.9 release.
2025-06-10	2.2.0 release.
2025-06-23	2.3.0 release.
2025-06-27	2.3.1 release.
2025-07-02	2.3.2 release.
2025-07-21	2.4.0 release.
2025-07-23	2.5.0 release.
2025-08-06	2.5.1 release. Updated Get Started with the FortiCNAPP CLI on page 23.
2025-08-12	2.6.0 release.
2025-08-15	2.6.1 release.
2025-08-18	2.6.2 release.
2025-09-04	2.7.0 release.
2025-09-22	2.7.1 release.
2025-09-24	2.8.0 release.
2025-10-06	2.8.1 release.
2025-10-21	2.8.2 release.

Date	Change Description
2025-11-13	2.8.3 release.
2025-12-10	2.8.4 release.
2026-01-15	2.8.5 release.
2026-01-19	2.9.0 release.
2026-01-28	2.9.1 release.
2026-02-10	2.10.0 release.
2026-03-12	2.11.0 release.
2026-03-16	2.12.0 release.
2026-03-17	2.12.1 release.

Get Started with the FortiCNAPP CLI

The FortiCNAPP CLI is an open source project written in Golang and released as separate binaries for Linux, macOS, and Windows. All releases of the CLI are also published as Docker containers to Docker Hub for various platforms with the intended purpose of integrating with CI/CD automation pipelines.

FortiCNAPP as a platform provides a set of robust APIs for configuring accounts within the platform, as well as accessing data from accounts. The FortiCNAPP CLI provides an interface to those APIs with the goal of providing fast, accurate, and actionable insights into the platform.

Install the FortiCNAPP CLI

Bash (macOS/Linux)

```
curl https://raw.githubusercontent.com/lacework/go-sdk/main/cli/install.sh | bash
```

Powershell (Windows)

1. Open a PowerShell terminal and run the following command:

   ```
   Set-ExecutionPolicy Bypass -Scope Process -Force
   iex ((New-Object System.Net.WebClient).DownloadString
   ('https://raw.githubusercontent.com/lacework/go-sdk/main/cli/install.ps1'))
   ```

   The FortiCNAPP CLI is installed at `C:\ProgramData\Lacework\lacework.exe` and the system PATH environment variable is updated to include the FortiCNAPP CLI.
2. Open a new PowerShell terminal to read the updated system PATH and use the FortiCNAPP CLI.

Homebrew (macOS/Linux)

```
brew install lacework/tap/lacework-cli
```

For more details, see the Lacework Homebrew Tap.

Chocolatey (Windows):

```
choco install lacework-cli
```

For more details, see the Lacework CLI Chocolatey package.

Azure Cloud Shell

1. In Cloud Shell, switch to Bash.
2. Run the following commands:

   ```
   mkdir -p "$HOME"/bin
   curl https://raw.githubusercontent.com/lacework/go-sdk/main/cli/install.sh | bash -s -- -d
   "$HOME"/bin
   ```

3. Exit and reconnect to Cloud Shell.

Create API Key

The FortiCNAPP CLI requires an API key and secret to authenticate with FortiCNAPP. FortiCNAPP API Keys can be created by FortiCNAPP account administrators via the FortiCNAPP Console. For more information, go to API Access Keys and Tokens.

1. Log in to the FortiCNAPP Console.
2. Click **Settings > API keys**.
3. Click **Add New**.
4. Enter a name for the key and an optional description.
5. Click **Save**.
6. Click the **...** icon and then **Download** to save the API key file locally.

The contents of your API key contain a `keyId`, `secret`, `subAccount`, and `account`:

```
{
  "keyId": "ACCOUNT_ABCEF01234559B9B07114E834D8570F567C824039756E03",
  "secret": "_abc1234e243a645bcf173ef55b837c19",
  "subAccount": "myaccount",
  "account": "myaccount.lacework.net"
}
```

Configure the CLI

Use the `lacework configure` command to configure the FortiCNAPP CLI with the API Key downloaded from the previous step.

```
lacework configure -j /path/to/key.json
```

Example output:

```
Account: example
Access Key ID: EXAMPLE_1234567890ABCDE1EXAMPLE1EXAMPLE123456789EXAMPLE
Secret Access Key: **********************************

You are all set!
```

The `lacework configure` command generates a file named `.lacework.toml` inside your home directory (`$HOME/.lacework.toml`) with a single profile named `default`.

Enable Command Autocomplete

You can work faster and get help remembering commands by enabling command autocomplete for the FortiCNAPP CLI for your shell environment.

With autocomplete enabled, you can quickly complete CLI commands you have started to type by hitting the tab key. If there is more than one way to complete a command, hit the tab key twice to view all options.

The FortiCNAPP CLI supports several shell environments, including bash, powershell, and zsh.

To use autocomplete in a supported shell, you first need to enable it by running the autocomplete script. To see supported shell environments, run the following command:

```
lacework completion
```

For example, to enable completion in your current bash shell session, run the following command:

```
source <(lacework completion bash)
```

For complete instructions for bash and other supported environments, see the following topics:

- `lacework completion bash`
- `lacework completion fish`
- `lacework completion powershell`
- `lacework completion zsh`

For general information on the FortiCNAPP CLI command autocomplete feature, see lacework completion.

Multiple Profiles

You can add additional profiles that you can refer to with a name by specifying the `--profile` flag. The following example creates a profile named `prod`.

```
lacework configure --profile prod -j /path/to/key.json
```

Example output:

```
Account: prod.example
Access Key ID: PROD_1234567890ABCDE1EXAMPLE1EXAMPLE123456789EXAMPLE
Secret Access Key: **********************************

You are all set!
```

Then, when you run a command, you can specify a `--profile prod` and use the credentials and settings stored under that name.

```
lacework agent list --profile prod
```

If there is no `--profile` flag, the FortiCNAPP CLI defaults to the `default` profile.

To list all available profiles configured in the workstation use:

```
lacework configure list
```

Example output:

```
    PROFILE   |    ACCOUNT    |                         API KEY                          |
   API SECRET
--------------+---------------+----------------------------------------------------------+--------
----------------------------
    prod      | prod-account  | PRODACCT_0C66EF03A0694E16D3203E553C9B13E36E39239FB0FCEBF |
*****************************8520
    qa1       | qa1-account   | QA1ACCOT_038B1395C1B5B9BD1C5DEA849DF62FCB95D7697C58C4942 |
*****************************9ad8
    qa2       | qa2-account   | QA2ACCOT_0362BF5146FBE18A9CD0AB0259FBEE912EBB1A429A0A213 |
*****************************a3cb
  > default   | dev-account   | DEVACCOT_03C8910D0BDCDBD2AFD4355A1C5284104AAA2AE5253938C |
*****************************98f1
```

Switch Profiles

To switch between profiles configured into the config file `$HOME/.lacework.toml`, use the command.

```
lacework configure switch-profile <profile>
```

This is a global configuration for the FortiCNAPP CLI, which means that any new terminal continues to use the selected profile.

To switch back to the `default` profile.

```
lacework configure use default
```

The command `lacework configure use` is an alias to the `switch-profile` sub-command.

An alternative to temporarily switching to a different profile in your current terminal is to export the environment variable `LW_PROFILE=your-profile`

Organizational Accounts

An organization can contain multiple accounts so you can manage components such as alerts, resource groups, team members, and audit logs at a more granular level inside an organization. A team member may have access to multiple accounts and can easily switch between them.

To enroll your account in an organization, see Organization Enrollment Process.

Use the global flag `--subaccount` to switch to a different account inside your organizational account.

For example, having a `default` profile that has access to your primary account named `my-company`:

```
[default]
  account = "my-company"
  api_key = "my-api-key"
  api_secret = "my-api-secret"
  version = 2
```

To access your sub-account named `business-unit`, pass the flag `--subaccount business-unit` to any command.

There are two ways to set a sub-account persistently:

- Export the environment variable `LW_SUBACCOUNT="<YOUR_SUBACCOUNT>"`. This only makes the sub-account configuration persist for the active terminal.
- Reconfigure your profile with the command `lacework configure`. This command prompts you to select any sub-account that you have access to in your organizational account.

To list all accounts in your organization:

```
lacework account list
```

Output Formats

The FortiCNAPP CLI supports the following output formats:

- *Human-readable*: Default output that presents the information in a "human-readable" or "human-friendly" format, which is much easier to read, but not as useful for automation purposes.
- *JSON*: To switch the output of any command to be formatted as a JSON string, add the flag `--json`.

Some commands that have additional formats available:

- *PDF*: For compliance reports, it is possible to download a report in PDF format by adding the flag `--pdf`.
- *HTML*: For container vulnerability assessments, use the `--html` flag to render the assessment results in HTML format.

- *CSV*: For compliance reports and host vulnerability data, you can pass the flag `--csv` to switch the output to CSV format.

Environment Variables

Default configuration parameters found in the `.lacework.toml` may also be overridden by setting environment variables prefixed with `LW_`.

To override the `account`, `api_key`, and `api_secret` configurations:

- *Bash (macOS/Linux)*

```
export LW_ACCOUNT="<YOUR_ACCOUNT>"
export LW_API_KEY="<YOUR_API_KEY>"
export LW_API_SECRET="<YOUR_API_SECRET>"
```

- *Powershell (Windows)*

```
$env:LW_ACCOUNT = '<YOUR_ACCOUNT>'
$env:LW_API_KEY = '<YOUR_API_KEY>'
$env:LW_API_SECRET = '<YOUR_API_SECRET>'
```

For org admins only, to switch to a different sub-account permanently in your current terminal:

- *Bash (macOS/Linux)*

```
export LW_SUBACCOUNT=business-unit
```

- *Powershell (Windows)*

```
$env:LW_SUBACCOUNT = 'business-unit'
```

The following lists all environment variables that you can use to modify the operation of the FortiCNAPP CLI.

Environment Variable	Description
`LW_NOCOLOR=1`	turn off colors
`LW_NOCACHE=1`	turn off caching
`LW_DEBUG=1`	turn on debug logging
`LW_JSON=1`	switch commands output from human-readable to JSON format
`LW_NONINTERACTIVE=1`	disable interactive progress bars (i.e. spinners)
`LW_UPDATES_DISABLE=1`	disable daily version checks
`LW_TELEMETRY_DISABLE=1`	disable sending telemetry data
`LW_PROFILE="<name>"`	switch between profiles configured at `~/.lacework.toml`
`LW_ACCOUNT="<account>"`	account subdomain of URL (i.e. `<ACCOUNT>.lacework.net`)

Environment Variable	Description
`LW_API_KEY="<key>"`	API access key id
`LW_API_SECRET="<secret>"`	API secret access key
`LW_SUBACCOUNT="<sub-account>"`	sub-account name inside your organization (org admins only)

Code Security component installation

The FortiCNAPP Code Security suite uses a Cloud Development Kit (CDK) model to package and upgrade IaC and SCA components. The `lacework component <command> <component>` command can be used to implement Code Security components. See Code Security in the FortiCNAPP Administration Guide for more information on IaC and SCA.

Available commands are included in the following table:

Command	Description
`install`	Install a new component. See lacework component install on page 144.
`list`	List all available components. See lacework component list on page 145.
`show`	Show details about a defined component. See lacework component show on page 146.
`uninstall`	Uninstall an existing component. See lacework component uninstall on page 147.
`update`	Update an existing component. See lacework component update on page 147.

Available components are included in the following table:

Component	Description
`preflight`	The preflight check for FortiCNAPP Cloud setup.
`remediate`	Isolate and remediate resources. See Lacework remediate on page 62.
`sca`	The Code Security Software Composition Analysis (SCA) offering. This component contains SAST, Secrets, SBOM, and licensing capabilities in addition to SCA scanning. See Software Composition Analysis (SCA) in the FortiCNAPP Administration Guide.
`component-example`	Review component descriptions.
`iac`	The Code Security Infrastructure-as-Code (IaC) offering. See Infrastructure-as-Code Security in the FortiCNAPP Administration Guide.

Example 1: Running FortiCNAPP component commands in the CLI

The following example demonstrates using the `lacework component list` command to review the installation status and current version of available components:

```
> lacework component list
     STATUS            NAME           VERSION                  DESCRIPTION
----------------+-------------------+---------+-------------------------------------------
  Not Installed   preflight           0.8.21    Preflight check for Cloud Setup
  Not Installed   remediate           0.6.4     A tool to isolate and remediate resources
  Not Installed   sca                 0.1.66    Software Component Analysis
  Not Installed   component-example   0.9.7     Component description
  Not Installed   iac                 0.10.31   Infrastructure as Code (IaC) scanner
```

See lacework component list on page 145.

Example 2: Installing the Code Security IaC component

The following example demonstrates installing the FortiCNAPP Code Security IaC offering:

```
> lacework component install iac
 [√] Component iac found
 [√] Component iac staged
 [√] Component signature verified
 [√] Component version 0.10.31 installed
 [√] Component configured

Installation completed.

IAC component successfully installed. You can start using scanning your code, see:

  lacework iac --help
```

See lacework component install on page 144.

Example 3: Installing the Code Security SCA component

The following example demonstrates installing the FortiCNAPP Code Security SCA and SAST offerings through the `sca` component:

```
> lacework component install sca
 [√] Component sca found
 [√] Component sca staged
 [√] Component signature verified
 [√] Component version 0.1.66 installed
 [✓] Component configured

Installation completed.

To check the version of SCA installed, run:

  lacework sca version
```

See lacework component install on page 144.

Create Policies with the CLI

PREVIEW FEATURE This article describes functionality that is currently in preview. Access to Azure, Google Cloud, and OCI datasources is in preview status.

This topic walks you through using the FortiCNAPP CLI to create a custom violation policy that detects unrestricted ingress to TCP port 445. Port 445 is conventionally used for SMB communication, and should not be open to external networks.

If you are new to the FortiCNAPP CLI, see Get Started to learn about installing and configuring the CLI.

This walkthrough follows a basic end-to-end workflow to create a custom policy. For additional options when creating queries and policies, see LQL Queries and Policies. For information about policy types, see Custom Policy Types.

Create a Query

What Datasources Are Available

The easiest way to learn about the LQL datasources is to discover the names of the datasources and then get details about the one you are interested in.

Run the command that corresponds to your cloud provider:

```
lacework query list-sources | grep AWS
```

```
lacework query list-sources | grep GCP
```

```
lacework query list-sources | grep AZURE
```

```
lacework query list-sources | grep OCI
```

What Fields Can I Use from a Datasource

In order to learn which fields to use in your query, run the `lacework query show-source` command for a description of the fields. For some datasources, you can run the `lacework query preview-source` command (not available for all datasources).

The following command shows the details for the `LW_CFG_AWS_EC2_SECURITY_GROUPS` datasource.

```
lacework query show-source LW_CFG_AWS_EC2_SECURITY_GROUPS
```

```
            DATASOURCE                      DESCRIPTION
---------------------------------+---------------------------------
  LW_CFG_AWS_EC2_SECURITY_GROUPS   Results from AWS EC2
                                   'describe-security-groups'

     FIELD NAME      DATA TYPE           DESCRIPTION
-------------------+-----------+---------------------------------
  BATCH_START_TIME   Timestamp   Beginning of time interval
  BATCH_END_TIME     Timestamp   End of time interval
  QUERY_START_TIME   Timestamp   Start time of query for this
                                 resource
  QUERY_END_TIME     Timestamp   End time of query for this
                                 resource
  ARN                String      ARN for the resource
  API_KEY            String      Key describing the API used to
                                 fetch data for this resource
  SERVICE            String      Service this resource belongs
                                 to
  ACCOUNT_ID         String      AWS Account ID
  ACCOUNT_ALIAS      String      User friendly alias for AWS
                                 Account
  RESOURCE_TYPE      String      Type of this resource
  RESOURCE_ID        String      Identifier for this resource
  RESOURCE_REGION    String      Region this resource belongs
                                 to
  RESOURCE_CONFIG    JSON        JSON Definition of this
                                 resource
  RESOURCE_TAGS      JSON        Tags associated with this
                                 resource
```

The `RESOURCE_CONFIG` field is frequently used in LQL. Because it is a JSON datasource, the LQL query must first convert the field using the `array_to_rows()` function. To know exactly which JSON fields you need, you can either read the cloud provider's API documentation, or write an LQL query to explore the full content before writing the actual policy.

Explore Datasources Using LQL

This example explores the `LW_CFG_AWS_EC2_SECURITY_GROUPS` datasource. Replace the datasource with `LW_CFG_GCP_COMPUTE_FIREWALL` or `LW_CFG_AZURE_NETWORK_NETWORKSECURITYGROUPS` if using Google Cloud or Azure, respectively.

1. Open your text editor, create a new file, and add the following content:

```
---
queryId: Explore_AWS_EC2_SECURITY_GROUPS
queryText: |-
  {
      source {
         LW_CFG_AWS_EC2_SECURITY_GROUPS
      }
      return {
         RESOURCE_CONFIG
      }
  }
```

2. Save the file as YAML with the filename Explore_AWS_EC2_SECURITY_GROUPS.yaml. Note the file's location.
3. In the FortiCNAPP CLI, run this command:

```
lacework query run -f <path_to>/Explore_AWS_EC2_SECURITY_GROUPS.yaml
```

```
{
 "RESOURCE_CONFIG": {
   "Description": "default VPC security group",
   "GroupId": "sg-000",
   "GroupName": "default",
   "IpPermissions": [
     {
       "IpProtocol": "-1",
       "IpRanges": [],
       "Ipv6Ranges": [],
       "PrefixListIds": [],
       "UserIdGroupPairs": [
         {
           "GroupId": "sg-000",
           "UserId": "111"
         }
       ]
     }
   ],
   "IpPermissionsEgress": [
     {
       "IpProtocol": "-1",
       "IpRanges": [
         {
           "CidrIp": "0.0.0.0/0"
         }
       ],
       "Ipv6Ranges": [],
       "PrefixListIds": [],
       "UserIdGroupPairs": []
     }
   ],
   "OwnerId": "111",
   "VpcId": "vpc-000"
 }
}
```

Create a Query

1. Open your text editor, create a new file, and add the content that corresponds to your cloud provider:

```
---
queryId: LW_Custom_UnrestrictedIngressToTCP445
queryText: |-
  {
      source {
          LW_CFG_AWS_EC2_SECURITY_GROUPS a,
          array_to_rows(a.RESOURCE_CONFIG:IpPermissions) as (ip_permissions),
          array_to_rows(ip_permissions:IpRanges) as (ip_ranges)
      }
      filter {
          ip_permissions:IpProtocol = 'tcp'
          and ip_permissions:FromPort = 445
          and ip_permissions:ToPort = 445
          and ip_ranges:CidrIp = '0.0.0.0/0'
      }
      return distinct {
          ACCOUNT_ALIAS,
          ACCOUNT_ID,
          ARN as RESOURCE_KEY,
          RESOURCE_REGION,
          RESOURCE_TYPE,
          SERVICE
      }
  }
```

```
---
queryId: LW_Custom_UnrestrictedIngressToTCP445
queryText: |-
  {
      source {
          LW_CFG_GCP_COMPUTE_FIREWALL firewall,
          array_to_rows(firewall.RESOURCE_CONFIG:allowed) as (allowed),
          array_to_rows(allowed:ports) as (ports),
          array_to_rows(firewall.RESOURCE_CONFIG:sourceRanges) as (ranges)
      }
      filter {
          RESOURCE_CONFIG:direction = 'INGRESS'
          and allowed:IPProtocol = 'tcp'
          and ports = '445'
          and ranges = '0.0.0.0/0'
      }
      return distinct {
          ORGANIZATION_ID,
          PROJECT_NUMBER,
          PROJECT_ID,
          FOLDER_IDS,
          URN as RESOURCE_KEY,
          RESOURCE_REGION,
          RESOURCE_TYPE,
          SERVICE
      }
  }
```

```
---
queryId: LW_Custom_UnrestrictedIngressToTCP445
queryText: |-
  {
      source {
          LW_CFG_AZURE_NETWORK_NETWORKSECURITYGROUPS a,
          array_to_rows(a.RESOURCE_CONFIG:securityRules) as (rules)
      }
      filter {
          rules:"properties".access = 'Allow'
          and rules:"properties".direction = 'Inbound'
          and rules:"properties".protocol = 'Tcp'
          and rules:"properties".destinationPortRange = '445'
          and rules:"properties".sourceAddressPrefix = '*'
      }
      return distinct {
          TENANT_ID,
          TENANT_NAME,
          SUBSCRIPTION_ID,
          SUBSCRIPTION_NAME,
          URN as RESOURCE_KEY,
          RESOURCE_REGION,
          RESOURCE_TYPE
      }
  }
```

```
---
queryId: LW_Custom_UnrestrictedIngressToTCP445
queryText: |-
   {
      source {
            LW_CFG_OCI_NETWORK_NETWORK_SECURITY_GROUPS
      }
      filter {
            RESOURCE_ID in {
               source {
                  LW_CFG_OCI_NETWORK_NETWORK_SECURITY_GROUP_RULES
               }
               filter {
                  RESOURCE_CONFIG:direction = 'INGRESS'
                  and RESOURCE_CONFIG:"source" = '0.0.0.0/0'
                  and RESOURCE_CONFIG:protocol = '6'
                  and RESOURCE_CONFIG:tcp_options.destination_port_range.min <= 445
                  and RESOURCE_CONFIG:tcp_options.destination_port_range.max >= 445
               }
               return distinct {
                  RESOURCE_ID
               }
            }
      }
      return distinct {
            COMPARTMENT_ID,
            RESOURCE_KEY,
            RESOURCE_REGION,
            RESOURCE_TYPE,
            SERVICE,
      }
   }
```

2. Save the file as YAML with the filename LW_Custom_UnrestrictedIngressToTCP445.yaml. Note the file's location.
3. In the FortiCNAPP CLI, run this command:

```
lacework query create -f <path_to>/LW_Custom_UnrestrictedIngressToTCP445.yaml
```

```
The query LW_Custom_UnrestrictedIngressToTCP445 was created.
```

Test Using Collected Data

In the FortiCNAPP CLI, run this command:

```
lacework query run LW_Custom_UnrestrictedIngressToTCP445
```

```
[
  {
     "ACCOUNT_ALIAS": "",
     "ACCOUNT_ID": "aaa",
     "RESOURCE_KEY": "arn:aws:ec2:us-east-2:aaa:security-group/sg-bbb",
     "RESOURCE_REGION": "us-east-2",
     "RESOURCE_TYPE": "ec2:security-group",
     "SERVICE": "ec2"
  }
]
```

```
[
  {
     "FOLDER_IDS": [],
     "ORGANIZATION_ID": "aaa",
     "PROJECT_ID": "ccc",
     "PROJECT_NUMBER": "bbb",
     "RESOURCE_KEY": "gcp:bbb://compute.googleapis.com/projects/ccc/global/firewalls/ddd",
     "RESOURCE_REGION": "global",
     "RESOURCE_TYPE": "compute.googleapis.com/Firewall",
     "SERVICE": "compute"
  }
]
```

```
[
  {
     "RESOURCE_KEY":
"/subscriptions/aaa/resourceGroups/bbb/providers/Microsoft.Network/networkSecurityGroups/ddd",
     "RESOURCE_REGION": "westus",
     "RESOURCE_TYPE": "microsoft.network/networksecuritygroups",
     "SUBSCRIPTION_ID": "aaa",
     "SUBSCRIPTION_NAME": "eee",
     "TENANT_ID": "ccc",
     "TENANT_NAME": "fff"
  }
]
```

```
[
  {
     "COMPARTMENT_ID": "ocid1.tenancy.oc1..aaa",
     "RESOURCE_KEY": "ocid1.networksecuritygroup.oc1.us-sanjose-1.aaa",
     "RESOURCE_REGION": "us-sanjose-1",
     "RESOURCE_TYPE": "nsg",
     "SERVICE": "network"
  }
]
```

Create a Policy

1. Open your text editor, create a new file, and add the content that corresponds to your cloud provider:

```
---
title: Security Groups Should Not Allow Unrestricted Ingress to TCP Port 445
enabled: false
policyType: Violation
alertEnabled: false
alertProfile: LW_CFG_AWS_DEFAULT_PROFILE.CFG_AWS_Violation
evalFrequency: Daily
queryId: LW_Custom_UnrestrictedIngressToTCP445
severity: high
description: Security groups should not allow unrestricted ingress to TCP port 445
remediation: Policy remediation
```

```
---
title: Security Groups Should Not Allow Unrestricted Ingress to TCP Port 445
enabled: false
policyType: Violation
alertEnabled: false
```

```
alertProfile: LW_CFG_GCP_DEFAULT_PROFILE.Violation
evalFrequency: Daily
queryId: LW_Custom_UnrestrictedIngressToTCP445
severity: high
description: Security groups should not allow unrestricted ingress to TCP port 445
remediation: Policy remediation
```

```
---
title: Network Security Groups Should Not Allow Unrestricted Ingress to TCP Port 445
enabled: false
policyType: Violation
alertEnabled: false
alertProfile: LW_CFG_AZURE_DEFAULT_PROFILE.Violation
evalFrequency: Daily
queryId: LW_Custom_UnrestrictedIngressToTCP445
severity: high
description: Network security groups should not allow unrestricted ingress to TCP port 445
remediation: Policy remediation
```

```
---
title: Network Security Groups Should Not Allow Unrestricted Ingress to TCP Port 445
enabled: false
policyType: Violation
alertEnabled: false
alertProfile: LW_CFG_OCI_DEFAULT_PROFILE.Violation
evalFrequency: Daily
queryId: LW_Custom_UnrestrictedIngressToTCP445
severity: high
description: Network security groups should not allow unrestricted ingress to TCP port 445
remediation: Policy remediation
```

The fields in the policy definition are:

- `title`: Customize the event title.
- `enabled`: Enable or disable the policy (`true|false`).
- `policyType`: Enter `Violation` as the `policyType`.
- `alertEnabled`: Enable or disable alerts (`true|false`).
- `alertProfile`: Provide the `alertProfile` and `alert` template name within the alert profile. It follows the format `alertProfileId.alert_template_name`.
- `evalFrequency`: Optional. Set the `evalFrequency` (`Hourly|Daily`).
- `queryId`: Provide the `queryID`. It must match the ID of the query you want to use.
- `severity`: Set the desired severity (`critical|high|medium|low|info`).
- `description`: Customize the description to display.
- `remediation`: Customize the remediation message to display.

2. Save the file as YAML with the filename UnrestrictedIngressToTCP445.yaml. Note the file's location.
3. In the FortiCNAPP CLI, run this command:

```
lacework policy create -f <path_to>/UnrestrictedIngressToTCP445.yaml
```

```
The policy <policy_name> was created.
```

Enable Alerts

The policy you just created is not enabled and does not yet raise alerts when violations occur. The following enables alerts for the policy.

1. In your text editor, open the UnrestrictedIngressToTCP445.yaml file and update these fields:
   - `enabled: false` to `enabled: true`
   - `alertEnabled: false` to `alertEnabled: true`
2. Save your changes.
3. In the FortiCNAPP CLI, run this command:

   ```
   lacework policy update -f <path_to>/UnrestrictedIngressToTCP445.yaml
   ```

   ```
   The policy <policy_name> was updated.
   ```

Time Format

The FortiCNAPP API requires use of RFC 3339 format when referencing dates and times. The FortiCNAPP CLI, however, adds features that make specifying dates and times easier and more flexible: relative time specifiers and natural time ranges.

Relative Time Specifiers for LQL Queries

Relative times allow you to represent time values dynamically, using specifiers that represent an offset from the current time. For instance, a relative time of `-24h` produces a date/time that is 24 hours less the current time. Relative times can also snap to a particular time. For instance, a relative time of `@d` would represent the start of the current day.

For example, the following command specifies a time range (using a start and end time) that represents the previous day:

```
lacework query run [query_id] --start -1d@d --end @d
```

A relative time has three components:

- A signed (+/-) integer
- A relative time unit
- A relative time snap

FortiCNAPP supports the following relative time units:

- y - year
- mon - month
- w - week
- d - day
- h - hour
- m - minute
- s - second

Additional considerations include:

- To represent the current time, you can specify either `now` or `+0s`.
- When specifying an integer and relative time unit, snaps are optional.
- When specifying a snap, the integer and relative time unit are optional. For instance, `@d` is actually interpreted as `+0s@d`.

Natural Time Ranges

Natural time ranges allow you to represent time range values using natural language in CLI commands and LQL queries. For instance, a natural time range of `yesterday` represents a relative start time of `-1d@d` and a relative end time of `@d`.

For example, the following command specifies a time range of this month:

```
lacework query run --range "this month"
```

A natural time has three components:

- An adjective
- A positive number (only when using the last adjective)
- The full text representation of a relative time unit (i.e., year/years)

FortiCNAPP supports the following adjectives (disambiguating previous and last by design):

- this/current
- previous
- last

Additional considerations include:

- `last` implies "in the last". So last week reads as "in the last week" and represents a start time of `-1w` and an end time of `now`.
- `previous` always snaps. So "previous week" represents a start time of `-1w@w` and an end time of `@w`.
- `yesterday` is a valid natural time and is equivalent to previous day.
- `today` is a valid natural time and is equivalent to this day or current day.

Agent Management

To analyze application, host, and user behavior, FortiCNAPP uses a lightweight agent, which securely forwards collected metadata to the FortiCNAPP platform for analysis. The agent requires minimal system resources and runs on most Linux distributions.

Install an Agent

Use the command `lacework agent install <[user@]host[:port]>` for single-host installation of the FortiCNAPP agent via Secure Shell (SSH). When this command is executed without any additional flag, an interactive prompt will be launched to help gather the necessary authentication information to access the remote host.

For a complete list of supported installation methods, see Agent Install Options.

To authenticate to the remote host with a username and password.

```
lacework agent install <host> --ssh_username <your-user> --ssh_password <secret>
```

To authenticate to the remote host with an identity file instead.

```
lacework agent install <user@host> -i /path/to/your/key
```

To provide an agent access token of your choice, use the command `lacework agent token list`, select a token and pass it to the `--token` flag.

To authenticate to the remote host on a non-standard SSH port use the '--ssh_port' flag or pass it directly via the argument.

```
lacework agent install <user@host:port>
```

To bypass the question to add unknown host keys to the `~/.ssh/known_hosts` file, use the flag `--trust_host_key`.

List Agents

List all hosts that have a running agent in your environment using the command.

```
lacework agent list
```

You can use key:value pairs to filter the list of hosts with the --filter flag.

```
lacework agent list --filter 'os:Amazon Linux' --filter 'tags.VpcId:vpc-72225916'
```

The value can be a regular expression such as hostname:db-server.*

Agent Access Tokens

To list all agent access tokens:

```
lacework agent token list
```

Agent tokens should be treated as secret and not published. A token uniquely identifies a FortiCNAPP customer. If you suspect your token has been publicly exposed or compromised, generate a new token, update the new token on all machines using the old token. When complete, the old token can safely be disabled without interrupting FortiCNAPP services.

To create a new agent access token:

```
lacework agent token create <name> [description]
```

The [description] is an optional argument.

You can use the agent token name to logically separate your deployments, for example, by environment types (QA, Dev, etc.) or system types (CentOS, RHEL, etc.).

To show details about an agent access token:

```
lacework agent token show <token>
```

By design, agent tokens cannot be deleted.

To disable an agent access token:

```
lacework agent token update <token> --disable
```

To enable an agent access token:

```
lacework agent token update <token> --enable
```

You can also update the name and/or description of any agent access token with the command:

```
lacework agent token update <token> --name dev --description "k8s deployment for dev env"
```

Compliance Reports

The FortiCNAPP cloud security platform provides continuous compliance monitoring against cloud security best practices and compliance standards such as CIS, PCI DSS, SOC II, and HIPAA benchmark standards.

Compliance reports run automatically within the FortiCNAPP platform at a time defined by the Resource Management Collection Schedule. You can use the `lacework compliance` command to interact with the three major cloud providers we support, AWS, Google Cloud, and Azure Cloud.

To integrate code to onboard one or more cloud accounts.

```
lacework generate cloud-account [cloud]
```

To configure the integration via the FortiCNAPP Console, log in to your account at:

```
https://<ACCOUNT>.lacework.net
```

Then navigate to *Settings* > *Integrations* > *Cloud Accounts*.

Compliance for AWS

List Configured Accounts

To list all AWS accounts configured in your account.

```
lacework compliance aws list
```

Get Compliance Report for AWS

To visualize a compliance report for an AWS account.

```
lacework compliance aws get-report <account_id>
```

- Extend the details of a compliance report by providing the `--details` flag
- Download the report in PDF format by specifying the `--pdf` flag
- Output the report in CSV format with the `--csv` flag
- Filter the recommendations table with `--category`, `--severity`, `--status`, `--service` flags
- To work with a different report type, use the `--type` flag (default report type is CIS)

To use filtering flags on a compliance report.

```
lacework compliance aws get-report <account_id> --category s3 --status non-compliant --severity
high
```

To show recommendation details and affected resources for a recommendation ID.

```
lacework compliance aws get-report <account_id> [recommendation_id]
```

Compliance for Google Cloud

List Configured Organizations/Projects

To list all GCP organizations and projects configured in your account.

```
lacework compliance gcp list
```

When integrating single GCP projects, this command displays the organization ID as `n/a`, which must be used as a parameter in subsequent commands as the `<organization_id>`.

Get Compliance Report for Google Cloud

To visualize a compliance report for a GCP project.

```
lacework compliance gcp get-report <organization_id> <project_id>
```

- Extend the details of a compliance report by providing the `--details` flag
- Download the report in PDF format by specifying the `--pdf` flag
- Output the report in CSV format with the `--csv` flag
- Filter the recommendations table with `--category`, `--severity`, `--status`, `--service` flags
- To work with a different report type, use the `--type` flag (default report type is CIS)

To use filtering flags on a compliance report.

```
lacework compliance gcp get-report <organization_id> <project_id> --category networking --status
non-compliant --severity high
```

To show recommendation details and affected resources for a recommendation id.

```
lacework compliance gcp get-report <organization_id> <project_id> [recommendation_id]
```

Compliance for Azure Cloud

List Configured Tenants/Subscriptions

To list all Azure tenants and subscriptions configured in your account.

```
lacework compliance azure list
```

Get Compliance Report for Azure

To visualize a compliance report for an Azure subscription.

```
lacework compliance azure get-report <tenant_id> <subscription_id>
```

- Extend the details of a compliance report by providing the `--details` flag
- Download the report in PDF format by specifying the `--pdf` flag
- Output the report in CSV format with the `--csv` flag
- Filter the recommendations table with `--category`, `--severity`, `--status`, `--service` flags
- To work with a different report type, use the `--type` flag (default report type is CIS)

To use filtering flags on a compliance report.

```
lacework compliance azure get-report <tenant_id> <subscription_id> --category storage --status
non-compliant --severity high
```

To show recommendation details and affected resources for a recommendation id.

```
lacework compliance azure get-report <tenant_id> <subscription_id> [recommendation_id]
```

LQL Queries

The Lacework Query Language (LQL) is a SQL-like query language for specifying the selection, filtering, and manipulation of FortiCNAPP data. Queries let you interactively request information from specified curated datasources. Queries have a defined structure for authoring detections.

For complete information on LQL, including examples and the list of datasources you can query, see the LQL documentation (login required).

Basic FortiCNAPP Query Commands

FortiCNAPP offers a set of default LQL queries that are available in your account.

To view all LQL queries in your FortiCNAPP account.

```
lacework query ls
```

To show a query.

```
lacework query show <query_id>
```

To delete a query.

```
lacework query delete <query_id>
```

LQL syntax may change.

Create a Query

There are multiple ways you can create a query:

- Type the query into your default editor (via `$EDITOR`)
- Pipe the query to the FortiCNAPP CLI command (via `$STDIN`)
- From a local file on disk using the flag `--file`
- From a URL using the flag `--url`

There are two formats you can use to define a query:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and create a new query.

```
lacework lql create
```

The following example checks for unrestricted ingress to TCP port 445.:

```
---
queryId: LW_Custom_UnrestrictedIngressToTCP445
queryText: |-
  {
     source {
         LW_CFG_AWS_EC2_SECURITY_GROUPS a,
         array_to_rows(a.RESOURCE_CONFIG:IpPermissions) as (ip_permissions),
         array_to_rows(ip_permissions:IpRanges) as (ip_ranges)
     }
     filter {
         ip_permissions:IpProtocol = 'tcp'
         and ip_permissions:FromPort = 445
         and ip_permissions:ToPort = 445
         and ip_ranges:CidrIp = '0.0.0.0/0'
     }
     return distinct {
         ACCOUNT_ALIAS,
         ACCOUNT_ID,
         ARN as RESOURCE_KEY,
         RESOURCE_REGION,
         RESOURCE_TYPE,
         SERVICE
     }
  }
```

This query specifies an identifier of `LW_Custom_UnrestrictedIngressToTCP445`, and identifies AWS EC2 security groups with unrestricted access to TCP port 445. The `queryText` is expressed in Lacework Query Language (LQL) syntax, which is delimited by '`{ }`' and contains three sections:

- Source data is specified in the 'source' clause. The source of data is the `LW_CFG_AWS_EC2_SECURITY_GROUPS` datasource. LQL queries generally refer to other datasources, and customizable policies always target a suitable datasource.
- Records of interest are specified by the `filter` clause. In the example, the records available in `LW_CFG_AWS_EC2_SECURITY_GROUPS` are filtered for those whose IP protocol is `tcp`, whose from and to port is `445`, and CidrIP is `0.0.0.0/0`. The syntax for this filtering expression strongly resembles SQL.
- The fields this query exposes are listed in the `return` clause. Because there may be unwanted duplicates among result records when FortiCNAPP composes them from just these columns, the distinct modifier is added. This behaves like a SQL `SELECT DISTINCT`. Each returned column in this case is just a field that is present in `LW_CFG_AWS_EC2_SECURITY_GROUPS`, but we can compose results by manipulating strings, dates, JSON and numbers as well.

The resulting dataset is shaped like a table. The table's columns are named with the names of the columns selected. If desired, you could alias them to other names as well.

Run a Query

To run an LQL query via editor.

```
lacework query run --range today
```

Run a query via ID (uses active profile):

```
lacework query run MyQuery --start "-1w@w" --end "@w"
```

Start and end times are required to run a query.

- Specify start and end times in one of the following formats:
  - A relative time specifier
  - RFC3339 date and time
  - Epoch time in milliseconds
- Specify start and end times in one of the following ways:
  - As `StartTimeRange` and `EndTimeRange` in the `ParamInfo` block within the query
  - As start_time_range and end_time_range if specifying JSON
  - As `--start` and `--end` flags
- Start and end time precedence:
  - CLI flags take precedence over JSON specifications
  - JSON specifications take precedence over `ParamInfo` specifications

Update a Query

There are multiple ways you can update a query:

- Type the query into your default editor (via `$EDITOR`)
- Pass the query ID to load it into your default editor
- From a local file on disk using the flag `--file`
- From a URL using the flag `--url`

There are two formats you can use to define a query:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and update a query.

```
lacework query update <query_id>
```

Explore Data Sources

Several questions arise as you begin to write LQL queries.

What are the data sources I can query?

```
lacework query list-sources
```

What fields within the data source are available to me?

```
lacework query show-source <datasource_id>
```

How can I see a sample event from a specified datasource?

```
lacework query preview-source <datasource_id>
```

CLI Policy Management

Policies add annotated metadata to queries for improving the context of alerts, reports, and information displayed in the FortiCNAPP Console.

Policies also facilitate the scheduled execution of a FortiCNAPP query.

Queries let you interactively request information from specified curated datasources. Queries have a defined structure for authoring detections.

FortiCNAPP offers a set of default LQL policies that are available in your account.

Limitations:

- The maximum number of records that each policy will return is 1000
- The maximum number of API calls is 120 per hour for on-demand LQL query executions

To view all the policies in your FortiCNAPP account.

```
lacework policy ls
```

- To show only enabled policies, use the `--enabled` flag
- To show only policies with the alert functionality enabled, use the `--alert_enabled` flag
- To filter policies by severity threshold (critical, high, medium, low, info), use the `--severity` flag
- To filter policies by tag, use the `--tag` flag

To list all tags associated with policies.

```
lacework policy list-tags
```

To view more details about a single policy.

```
lacework policy show <policy_id>
```

To view the LQL query associated with the policy, use the query ID.

```
lacework query show <query_id>
```

LQL syntax may change.

To delete a policy.

```
lacework policy delete <policy_id>
```

Create a Policy

There are multiple ways you can create a policy:

- Type the policy into your default editor (via `$EDITOR`)
- Pipe the policy to the FortiCNAPP CLI command (via `$STDIN`)
- From a local file on disk using the flag `--file`
- From a URL using the flag `--url`

There are two formats you can use to define a policy:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and create a new policy.

```
lacework policy create
```

The following attributes are required:

```
---
title: My Policy
enabled: false
policyType: Violation
alertEnabled: false
alertProfile: Alert_Profile_ID.Alert_Template_Name
evalFrequency: Daily
queryId: MyQuery
severity: high
description: My Policy Description
remediation: My Policy Remediation
```

To view all LQL queries in your FortiCNAPP account.

```
lacework query ls
```

For more information about queries, see LQL Queries.

Update a Policy

There are multiple ways you can update a policy:

- Type the policy into your default editor (via `$EDITOR`)
- Pipe the policy to the FortiCNAPP CLI command (via `$STDIN`)
- From a local file on disk using the flag `--file`
- From a URL using the flag `--url`

There are two formats you can use to define a policy:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor to update a policy.

```
lacework policy update <policy_id>
```

A policy identifier specifed via command argument always takes precedence over a policy identifer specified via payload.

Alert Insights

The command `lacework alert` helps you perform initial discovery and analysis of alerts happening in your FortiCNAPP account.

You can quickly see the list of all the alerts from the last 7 days in your account with their severity:

```
lacework alert list
```

This command is limited to displaying 7 days of data.

To filter alerts by a time period:

- Specify a start time with the flag `--start`.
- Specify both start and end times with the flags `--start` and `--end`.

To show all the alerts from a specific start time that has severity medium and above (critical, high, and medium):

```
lacework alert list --start 2020-08-26T23:28:29Z --severity medium
```

Time constraint: The start time must be within the last 92 days. The difference between start and end time should not be greater than 7 days.

There are different types of alert details that can be shown to assist with alert investigation. These types are referred to as alert detail scopes.

The following alert detail scopes are available:

- Details (default)
- Investigation
- Events
- RelatedAlerts
- Integrations
- Timeline

To drill into an alert and show its details with the default scope:

```
lacework alert show <alert_id>
```

View an alert's details with the timeline scope:

```
lacework alert show <alert_id> --scope Timeline
```

To open an alert in the FortiCNAPP Console for further investigation:

```
lacework alert open <alert_id>
```

Host Vulnerability

FortiCNAPP provides the ability to assess, identify, and report vulnerabilities found on Linux hosts within your environment. This means you can identify and take action on software vulnerabilities in your VMs and manage that risk proactively.

The FortiCNAPP CLI provides the `lacework vulnerability host` command to retrieve data on host vulnerability assessments with the intention of providing fast, accurate, and actionable data via FortiCNAPP's APIs. This includes the ability to list all CVEs found on hosts in your environment, search for hosts in your environment that have a specific CVE, show the assessment for a specific host, and the ability to submit a `package-manifest.json` for on-demand scanning of vulnerabilities.

The `lacework vulnerability host` command is not supported on Windows Server hosts.

To list the CVEs found in the hosts in your environment:

```
lacework vulnerability host list-cves
```

Additionally, you can filter results with the following flags:

- `--active` displays only vulnerabilities that are active within your environment
- `--fixable` displays only vulnerabilities with fixes
- `--packages` modifies the output format to show a list of packages with CVE count

To list the hosts that contain a specific CVE in your environment:

```
lacework vulnerability host list-hosts <cve_id>
```

To show the results of a host vulnerability assessment:

```
lacework vulnerability host show-assessment <machine_id>
```

Additionally, you can filter results with the following flags:

- `--active` displays only vulnerabilities that are active within your environment
- `--fixable` displays only vulnerabilities with fixes
- `--packages` modifies the output format to show a list of packages with CVE count

On-demand Assessment of Package Manifest

To request an on-demand host vulnerability assessment of your software packages to determine if the packages contain any common vulnerabilities and exposures:

```
lacework vulnerability host scan-pkg-manifest '{
    "osPkgInfoList": [
        {
            "os":"Ubuntu",
            "osVer":"18.04",
            "pkg": "openssl",
            "pkgVer": "1.1.1-1ubuntu2.1~18.04.5"
        }
    ]
}'
```

- Only packages managed by a package manager for supported OS's are reported.
- Calls to this operation are rate limited to 10 calls per hour, per access key.
- This operation is limited to 10k packages per command execution.

You can use the FortiCNAPP CLI to generate a package-manifest formatted and ready to be submitted for evaluation.

```
lacework vulnerability host generate-pkg-manifest
```

This command doesn't require any CLI configuration because it is meant to be executed on a running host.

To automatically generate a package manifest from the local host and send it directly to the FortiCNAPP platform for evaluation.

```
lacework vulnerability host scan-pkg-manifest --local
```

For a guided tutorial that shows how to build base images from code that are free of vulnerabilities and validated with FortiCNAPP’s host vulnerability scanning, see the blog post Up and Running with Lacework and Hashicorp Packer.

Container Vulnerability

The FortiCNAPP Platform provides the capability to scan container images for vulnerabilities at both build time and runtime. The FortiCNAPP CLI provides the `lacework vulnerability container` sub-command with a number of capabilities to retrieve data about container vulnerability assessments. This command is intended for use by individuals or teams responsible for tracking and remediating vulnerabilities. It provides data to help with prioritizing vulnerability remediation, with the ability to sort assessments by what is actively running in the environment and to filter on vulnerabilities that have available fixes.

To view all container vulnerability assessments for your FortiCNAPP account for the last 24 hours (default):

```
lacework vulnerability container list-assessments
```

Additionally, you can filter results with the following flags:

- `--fixable` displays only vulnerabilities with fixes
- `--repository` displays assessments for the specific repository
  Note: You may pass this flag multiple times to filter on multiple repositories
- `--registry` displays assessments for the specific registry
- `--start` specifies the start of the time range in UTC (format: `yyyy-MM-ddTHH:mm:ssZ`)
- `--end` specifies the end of the time range in UTC (format: `yyyy-MM-ddTHH:mm:ssZ`)
- `--range` natural time range

You can specify different start and end times in one of the following formats:

- A relative time specifier
- RFC 3339 date and time
- Epoch time in milliseconds

To view all of the containers in your environment with vulnerabilities that have fixes.

```
lacework vulnerability container list-assessments --fixable
```

To request an on-demand container vulnerability scan.

```
lacework vulnerability container scan <registry> <repository> <tag|digest>
```

Where:

- `<registry>` is the container registry where the container image has been published
- `<repository>` is the repository name that contains the container image
- `<tag|digest>` could be, either a tag or an image digest to scan (digest format: `sha256:1ee...1d3b`)

Scans can take up to 15 minutes to return results.

The following is an example of integrating the `lacework vulnerability container` command into a CI pipeline. The specific example requests an on-demand container vulnerability scan and waits for the scan to complete (results will be displayed in the terminal):

```
lacework vulnerability container scan <registry> <repository> <tag|digest> --poll --noninteractive
```

The `--noninteractive` flag disables interactive progress bars.

When the flag `--poll` is specified, there are a few other flags you can use to modify the output of the assessment:

- `--fixable` displays only fixable vulnerabilities
- `--packages` modifies the output format to show a list of packages with CVE count
- `--html` generates a vulnerability assessment in HTML format
- `--fail_on_fixable` returns a non-zero exit code if the assessed container has fixable vulnerabilities
- `--fail_on_severity` allows you to specify a severity threshold to fail (return a non-zero exit code) if vulnerabilities are found
  (available severities are critical, high, medium, low, and info)

To view a specific container vulnerability assessment use the command.

```
lacework vulnerability container show-assessment <sha256:hash>
```

You can extend the details of a vulnerability assessment by providing the flag `--details`.

Additionally, there are a few more flags you can use to modify the output of the assessment:

- `--fixable` displays only fixable vulnerabilities
- `--packages` modifies the output format to show a list of packages with CVE count
- `--html` generates a vulnerability assessment in HTML format
- `--csv` outputs the assessment in CSV format
- `--fail_on_fixable` helps automated pipelines to fail if the assessed container has fixable vulnerabilities
- `--fail_on_severity` allows you to specify a severity threshold to fail if vulnerabilities are found (available severities are critical, high, medium, low, and info)

Generate Static HTML Vulnerability Assessment

To provide developers with clear, actionable, insights to understand and remediate vulnerabilities, the FortiCNAPP CLI has the ability to generate static HTML files of container vulnerability assessments.

Use the flag `--html` in the following commands:

- `lacework vulnerability container scan`
- `lacework vulnerability container show-assessment`

The result is a standalone HTML file that you can download and share with other teams without additional artifacts, and which looks exactly like the FortiCNAPP Console.

Lacework remediate

Install FortiCNAPP Remediate CLI Component

FortiCNAPP Remediate CLI is a tool for interacting with and resolving security alerts associated with resource compliance policy violations in your AWS infrastructure. It offers pre-built remediation templates that assess each alert and provide command-line remediation guidance for addressing specific issues.

Prerequisites

- FortiCNAPP CLI
- AWS CLI

We recommend using the latest CLI versions for both FortiCNAPP and AWS.

To access AWS resources, you need an AWS account, IAM credentials, and an IAM access key pair.

Compatibility - The FortiCNAPP `remediate` component does not support Windows operating system.

Install the FortiCNAPP Remediate CLI Component

To install the FortiCNAPP Remediate CLI component, run the following command:

```
lacework component install remediate
```

Upon successful completion, you should see the following text:

```
❯ lacework component install remediate
 [✓] Component remediate found
 [✓] Component remediate installed
 [√] Component signature verified
 [√] Component configured

Installation completed.

Having installed the 'remediate' component you unlocked a new command:

lacework remediate alert <alert_id>

You have also unlocked a new flag for existing commands like:

lacework alert list --fixable

Try running one of these commands!
```

Create an IAM Role with Least Privileges (Optional)

As a best practive, we recommend creating a dedicated IAM role that you assume whenever you run remediations. This role can be assigned the least set of permissions needed to carry out remediations.

To get the least-privilege policy, run the following command:

```
lacework remediate show-policy aws-iam
```

This prints out an AWS IAM policy with the smallest set of privileges necessary to carry out remediations.

The output of this command is likely to change as FortiCNAPP adds more remediations. Make sure that you use the latest output of the `show-policy` subcommand whenever you update the FortiCNAPP CLI or `lacework remediate` CLI component.

Next, create an AWS policy from the output:

```
lacework remediate show-policy aws-iam > remediation-policy.json
aws iam create-policy --policy-name lw_remediation --policy-document file://remediation-
policy.json
```

Create a role for remediation using the AWS CLI:

```
aws iam create-role --role-name lacework-remediation --assume-role-policy-document file://trust-
policy.json
```

where `trust-policy.json` is the policy that determines who can assume that role, for example:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:group/devops"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Attach the least privilege policy above to the role as follows:

```
aws iam attach-role-policy --role-name lacework-remediation --policy-arn
arn:aws:iam::123456789012:policy/lw_remediation
```

Configure the FortiCNAPP CLI to Assume a Role

You can specify which AWS profile to use and which role to assume when running `lacework remediate` by using the following command:

```
lacework remediate configure
```

Step through the questions as they appear:

```
Existing settings will be loaded as defaults.  To clear the value, simply input a space.
? AWS Profile: my-aws-profile
? AWS Role Name: lacework-remediation
```

Manage Alerts with CLI Commands

Use the commands listed in this page to effectively manage and resolve compliance alerts within your AWS infrastructure.

Discover Fixable Alerts

Get all fixable alerts by running the below command:

```
lacework alert list --fixable
```

Below is what you will see on the CLI:

```
❯ lacework alert list --fixable
  ALERT ID |       TYPE        |        NAME        | SEVERITY |        START TIME        |
 END TIME          | STATUS
-----------+-------------------+--------------------+----------+--------------------------+-------
-------------------+---------
    222172 | ComplianceChanged | Compliance changed | Medium   | 2023-01-30T08:00:00.000Z | 2023-
01-30T09:00:00.000Z | Open
```

```
Use 'lacework alert show <alert_id>' to see details for a specific alert.
Use 'lacework remediate alert <alert_id>' to fix a specific alert.
```

Fix an Alert

The remediation of an alert consists of several steps (initialization, resource selection, plan, apply, etc.). These steps are represented as flags for the `lacework remediate alert` command.

- *Initialization:*
  When issuing this command without any flags, FortiCNAPP will initialize (persist to disk) the remediation package. This prepares the remediation package without executing any automation (making changes).

  ```
  lacework remediate alert 12345
  ```

- *Planning:*
  Use `--plan` to generate an execution plan. This shows FortiCNAPP's actions to apply the remediation (without making changes).

  ```
  lacework remediate alert 12345 --plan
  ```

- *Resource Selection:*

  Use `--resources` to select and deselect resources.

  ```
  lacework remediate alert 12345 --resources
  ```

- *Application:*
  Use `--apply` to apply the remediation.

  ```
  lacework remediate alert 12345 --apply
  ```

- *Rollback:*

  Use `--rollback` to roll back the remediation (if supported).

  ```
  lacework remediate alert 12345 --rollback
  ```

Passing Credentials

The `remediate` component leverages the AWS CLI. When invoking the CLI, environment variables from the current environment are passed through to the component and the `aws` commands.

A specific profile from your AWS credential file can be specified using the `--aws-profile` flag. Below is an example command:

```
lacework remediate alert 12345 --apply --aws-profile security-engineering-admin
```

You can also use AWS Vault, a tool to securely store and access AWS credentials in a development environment. Below is an example command:

```
aws-vault exec security-engineering-admin -- lacework remediate alert 12345 --apply
```

Passing User Data

Use `--userdata` to specify user-defined variables inline.

```
lacework remediate alert 12345 --userdata user=someuser --userdata other=someother
```

Starting Over

Use `--clean` to start over (delete the local remediation package).

```
lacework remediate alert 12345 --clean
```

Setting a dedicated AWS IAM role

By default, AWS Remediations are set to assume the IAM role *lacework-remediation* before executing any actions. This ensures that all actions performed by the FortiCNAPP CLI tool can be traced back to the role. If you want to bypass the role assumption step, use the `--no-assume-role` flag.

```
lacework remediate alert 12345 --no-assume-role
```

You can also define the name of the role to assume with the `--aws-role-name` flag:

```
lacework remediate alert 12345 --aws-role-name my-iam-role
```

Least Privilege

You can access least privilege IAM policies through the `show-policy` subcommand. These policies include only the essential permissions required to carry out the activities specified in the comprehensive set of remediation templates.

Below is an example:

```
lacework remediate show-policy aws-iam
```

Configuration

The FortiCNAPP remediate component offers persistence mechanisms for frequently used flags, such as `aws-profile` and `aws-role-name`. These mechanisms ensure that these flags are retained and readily available when needed.

For more information about these mechanisms, run the following command:

```
lacework remediate configure -h
```

Remediation Templates

The templates in the following table are available for you to leverage:

ID	Title	Description
lwcustom-11	Remove World Writeable Policy for %{bucket}	This remediation template effectively eliminates S3 world-writable access by deleting the bucket policy.
lacework-global-37	Ensure IAM password policy requires minimum length of 14 or greater	To enhance account security and protect against brute force login attempts, ensure that the IAM password policy enforces a minimum length of 14 characters or more. Implementing a password complexity policy further strengthens the account's resilience.
lacework-global-38	Ensure IAM password policy prevents password reuse	IAM password policies can effectively prevent users from reusing the same password. It is strongly recommended to enforce a password policy that prohibits password reuse.
lacework-global-40	Delete non-compliant IAM access key for user	This policy deletes an IAM access key for a given user.
lacework-global-41	Ensure credentials unused for 45 days or greater are disabled	AWS IAM users have various types of credentials, including passwords and access keys, to access AWS resources. It is advisable to deactivate or remove any credentials that have remained unused for 45 days or more.
lacework-global-46	Create a support role to manage incidents with AWS Support	AWS offers a support center to provide technical assistance and incident response. This remediation action involves creating a least-privilege role specifically designed for managing incidents with AWS Support.
lacework-global-48	Create IAM Access Analyzer	This remediation template enables IAM Access analyzer for all regions.
lacework-global-49	Ensure MFA Delete is enabled on S3 buckets	Enabling MFA Delete on your sensitive and classified S3 bucket ensures that users are required to authenticate using two forms of authentication. Note: Enabling MFA Delete will also activate versioning for the bucket. Once versioning is enabled, it cannot be disabled, although you can choose to suspend versioning on the bucket.

ID	Title	Description
lacework-global-51	Enable EBS Encryption	This remediation template ensures the default enablement of EBS encryption across all regions.
lacework-global-66	Ensure a log metric filter and alarm exists for AWS Organization changes	This remediation template adds a metric filter alarm, SNS topic, and subscription for monitoring AWS Organization changes performed in the master AWS Account.
lacework-global-73	Deny HTTP requests to S3 buckets	This remediation template updates the S3 bucket policies to explicitly deny HTTP requests directed towards the respective S3 buckets.
lacework-global-76	Ensures AWS Config is enabled in all regions	This remediation template ensures AWS Config is enabled for all regions. In addition to the config recorder and config delivery channel, the config service requires an S3 bucket, SNS topic, and IAM role.
lacework-global-78	Enable Customer Managed KMS key rotation	This remediation template enables key rotation for Customer Managed KMS keys that do not have it already enabled.

Raw FortiCNAPP API

To access raw FortiCNAPP API endpoints, use the command `lacework api <method> <path>`.

This command is useful when troubleshooting the behavior of commands, or to access functionality that hasn't been exposed via commands.

By default, all requests access APIv2, that is, they are all prefixed with `/api/v2`.

To explore all available schemas from the FortiCNAPP API run.

```
lacework api get /schemas
```

For a complete list of available APIv2 endpoints, visit: https://YourAccount.lacework.net/api/v2/docs

Telemetry

Telemetry is additional information that helps us to better understand our customer's needs, diagnose issues, and deliver features that improve the customer experience.

The FortiCNAPP CLI collects telemetry, such as generic usage metrics, system and environment information, and errors.

For more details about types of telemetry collected, see Types of Information Collected.

The FortiCNAPP CLI does *not* collect personal information, such as IP address. It also does *not* collect sensitive information passed through arguments or flags, such as usernames and passwords.

Opt Out of Telemetry

To opt-out of telemetry for the FortiCNAPP CLI or Terraform, set the environment variable `LW_TELEMETRY_DISABLE` to true.

You must repeat the command for each new terminal or session or configure this environment variable globally.

Unix

```
export LW_TELEMETRY_DISABLE=1
```

Windows

```
setx LW_TELEMETRY_DISABLE 1
```

Types of Information Collected

- Usage information – Commands and subcommands that are run.
- Errors and diagnostic information – The status and duration of commands that are run, including exit codes, and failures when connecting to the FortiCNAPP API.
- System and environment information – The version, operating system (Windows, Linux, or macOS), installation method, and the environment in which the FortiCNAPP CLI is executed (for example, inside a CI system, or a terminal).

lacework

A tool to manage the Lacework cloud security platform.

Synopsis

The Lacework Command Line Interface is a tool that helps you manage the Lacework cloud security platform. Use it to manage compliance reports, external integrations, vulnerability scans, and other operations.

Start by configuring the Lacework CLI with the command:

```
lacework configure
```

This will prompt you for your Lacework account and a set of API access keys.

Options

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
  -h, --help                help for lacework
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework access-token - Generate temporary API access tokens
- lacework account - Manage accounts in an organization (org admins only)
- lacework agent - Manage Lacework agents
- lacework alert - Inspect and manage alerts

- lacework alert-channel - Manage alert channels
- lacework alert-profile - Manage alert profiles
- lacework alert-rule - Manage alert rules
- lacework api - Helper to call Lacework's API
- lacework cloud-account - Manage cloud accounts
- lacework completion - Generate the autocompletion script for the specified shell
- lacework compliance - Manage compliance reports
- lacework component - Manage components
- lacework configure - Configure the Lacework CLI
- lacework container-registry - Manage container registries
- lacework generate - Generate code to onboard your account
- lacework policy - Manage policies
- lacework policy-exception - Manage policy exceptions
- lacework query - Run and manage queries
- lacework report-rule - Manage report rules
- lacework resource-group - Manage resource groups
- lacework team-member - Manage team members
- lacework version - Print the Lacework CLI version
- lacework vulnerability - Container and host vulnerability assessments
- lacework vulnerability-exception - Manage vulnerability exceptions

lacework access-token

Generate temporary API access tokens

Synopsis

Generates a temporary API access token that can be used to access the Lacework API. The token will be valid for the duration that you specify.

```
lacework access-token [flags]
```

Options

```
  -d, --duration_seconds int   duration in seconds that the access token should remain valid
(default 3600)
  -h, --help                   help for access-token
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.

lacework account

Manage accounts in an organization (org admins only)

Synopsis

Manage accounts inside your Lacework organization.

An organization can contain multiple accounts so you can also manage components such as alerts, resource groups, team members, and audit logs at a more granular level inside an organization. A team member may have access to multiple accounts and can easily switch between them.

To enroll your Lacework account in an organization follow the documentation:

```
https://docs.lacework.com/organization-overview
```

Options

```
  -h, --help   help for account
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework account list - List all accounts

lacework account list

List all accounts

Synopsis

List all accounts in your organization.

```
lacework account list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework account - Manage accounts in an organization (org admins only)

lacework agent

Manage Lacework agents

Synopsis

Manage agents and agent access tokens in your account.

To analyze application, host, and user behavior, Lacework uses a lightweight agent, which securely forwards collected metadata to the Lacework cloud for analysis. The agent requires minimal system resources and runs on most 64-bit Linux distributions.

For a complete list of supported operating systems, visit:

```
https://docs.lacework.com/supported-operating-systems
```

Options

```
  -h, --help   help for agent
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework agent aws-install - Install the datacollector agent on all remote AWS hosts
- lacework agent gcp-install - Install the datacollector agent on all remote GCE hosts
- lacework agent install - Install the datacollector agent on a remote host
- lacework agent list - List all hosts with a running agent
- lacework agent token - Manage agent access tokens

lacework agent aws-install

Install the datacollector agent on all remote AWS hosts

Options

```
  -h, --help   help for aws-install
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug                 turn on debug logging
      --json                  switch commands output from human-readable to json format
      --nocache               turn off caching
      --nocolor               turn off colors
      --noninteractive        turn off interactive mode (disable spinners, prompts, etc.)
      --organization          access organization level data sets (org admins only)
  -p, --profile string        switch between profiles configured at ~/.lacework.toml
      --subaccount string     sub-account name inside your organization (org admins only)
```

See also

- lacework agent - Manage Lacework agents
- lacework agent aws-install ec2ic - Use EC2InstanceConnect to securely connect to EC2 instances
- lacework agent aws-install ec2ssh - Use SSH to securely connect to EC2 instances
- lacework agent aws-install ec2ssm - Use SSM to securely install the Lacework agent on EC2 instances

lacework agent aws-install ec2ic

Use EC2InstanceConnect to securely connect to EC2 instances

Synopsis

This command installs the agent on all EC2 instances in an AWS account using EC2InstanceConnect.

To filter by one or more regions:

```
lacework agent aws-install ec2ic --include_regions us-west-2,us-east-2
```

To filter by instance tag:

```
lacework agent aws-install ec2ic --tag TagName,TagValue
```

To filter by instance tag key:

```
lacework agent aws-install ec2ic --tag_key TagName
```

To explicitly specify the username for all SSH logins:

```
lacework agent aws-install ec2ic --ssh_username <your-user>
```

To provide an agent access token of your choice, use the command 'lacework agent token list', select a token and pass it to the '--token' flag. This flag must be selected if the '--noninteractive' flag is set.

```
lacework agent aws-install ec2ic --token <token>
```

To explicitly specify the server URL that the agent will connect to:

```
lacework agent aws-install ec2ic --server_url https://your.server.url.lacework.net
```

To specify an AWS credential profile other than 'default':

```
lacework agent aws-install ec2ic --credential_profile aws-profile-name
```

AWS credentials are read from the following environment variables: - AWS_ACCESS_KEY_ID - AWS_SECRET_ACCESS_KEY - AWS_SESSION_TOKEN (optional) - AWS_REGION (optional)

This command will only install the agent on hosts that are supported by EC2InstanceConnect. The supported AMI types are Amazon Linux 2 and Ubuntu 16.04 and later. There may also be a region restriction.

This command will automatically add hosts with successful connections to '~/.ssh/known_hosts' unless specified with '--trust_host_key=false'.

```
lacework agent aws-install ec2ic [flags]
```

Options

```
      --credential_profile string   AWS credential profile to use (default "default")
  -h, --help                        help for ec2ic
  -r, --include_regions strings     list of regions to filter on
  -n, --max_parallelism int         maximum number of workers executing AWS API calls, set if rate
limits are lower or higher than normal (default 50)
      --server_url https://         server URL that agents will talk to, prefixed with https://
(default "https://agent.lacework.net")
      --ssh_username string         username to login with
      --tag strings                 only install agents on infra with this tag
      --tag_key string              only install agents on infra with this tag key set
      --token string                agent access token
      --trust_host_key              automatically add host keys to the ~/.ssh/known_hosts file
(default true)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
```

```
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent aws-install - Install the datacollector agent on all remote AWS hosts

lacework agent aws-install ec2ssh

Use SSH to securely connect to EC2 instances

Synopsis

This command installs the agent on all EC2 instances in an AWS account using SSH.

To filter by one or more regions:

```
lacework agent aws-install ec2ssh --include_regions us-west-2,us-east-2
```

To filter by instance tag:

```
lacework agent aws-install ec2ssh --tag TagName,TagValue
```

To filter by instance tag key:

```
lacework agent aws-install ec2ssh --tag_key TagName
```

To provide an existing access token, use the '--token' flag. This flag is required when running non-interactively ('--noninteractive' flag). The interactive command 'lacework agent token list' can be used to query existing tokens.

```
lacework agent aws-install ec2ssh --token <token>
```

To explicitly specify the server URL that the agent will connect to:

```
lacework agent aws-install ec2ssh --server_url https://your.server.url.lacework.net
```

You will need to provide an SSH authentication method. This authentication method should work for all instances that your tag or region filters select. Instances must be routable from your local host.

To authenticate using username and password:

```
lacework agent aws-install ec2ssh --ssh_username <your-user> --ssh_password <secret>
```

To authenticate using an identity file:

```
lacework agent aws-install ec2ssh -i /path/to/your/key
```

To specify an AWS credential profile other than 'default':

```
lacework agent aws-install ec2ssh --credential_profile aws-profile-name
```

The environment should contain AWS credentials in the following variables: - AWS_ACCESS_KEY_ID - AWS_SECRET_ACCESS_KEY - AWS_SESSION_TOKEN (optional), - AWS_REGION (optional)

This command will automatically add hosts with successful connections to '~/.ssh/known_hosts' unless specified with '--trust_host_key=false'.

```
lacework agent aws-install ec2ssh [flags]
```

Options

```
      --credential_profile string   AWS credential profile to use (default "default")
  -h, --help                        help for ec2ssh
  -i, --identity_file string        identity (private key) for public key authentication (default
"~/.ssh/id_rsa")
  -r, --include_regions strings     list of regions to filter on
  -n, --max_parallelism int         maximum number of workers executing AWS API calls, set if rate
limits are lower or higher than normal (default 50)
      --server_url https://         server URL that agents will talk to, prefixed with https://
(default "https://agent.lacework.net")
      --ssh_password string         password for authentication
      --ssh_port int                port to connect to on the remote host (default 22)
      --ssh_username string         username to login with
      --tag strings                 only select instances with this tag
      --tag_key string              only install agents on infra with this tag key
      --token string                agent access token
      --trust_host_key              automatically add host keys to the ~/.ssh/known_hosts file
(default true)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
```

```
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent aws-install - Install the datacollector agent on all remote AWS hosts

lacework agent aws-install ec2ssm

Use SSM to securely install the Lacework agent on EC2 instances

Synopsis

This command installs the Lacework agent on all EC2 instances in an AWS account using SSM.

This command will create a role and instance profile with 'SSMManagedInstanceCore' attached and associate that instance profile with the target instances. If the target instances already have associated instance profiles, this command will not change their state. This command will teardown the IAM role and instance profile before exiting.

This command authenticates with AWS credentials from well-known locations on the user's machine. The principal associated with these credentials should have the 'AmazonEC2FullAccess', 'IAMFullAccess' and 'AmazonSSMFullAccess' policies attached.

Target instances must have the SSM agent installed and running for successful installation.

To skip IAM role / instance profile creation and instance profile association:

```
lacework agent aws-install ec2ssm --skip_iam_role_creation
```

To provide a preexisting IAM role with the 'SSMManagedInstanceCore' policy

```
lacework agent aws-install ec2ssm --iam_role_name IAMRoleName
```

To filter by one or more regions:

```
lacework agent aws-install ec2ssm --include_regions us-west-2,us-east-2
```

To filter by instance tag:

```
lacework agent aws-install ec2ssm --tag TagName,TagValue
```

To filter by instance tag key:

```
lacework agent aws-install ec2ssm --tag_key TagName
```

To provide an agent access token of your choice, use the command 'lacework agent token list', select a token and pass it to the '--token' flag. This flag must be selected if the '--noninteractive' flag is set.

```
lacework agent aws-install ec2ssm --token <token>
```

To explicitly specify the server URL that the agent will connect to:

```
lacework agent aws-install ec2ssm --server_url https://your.server.url.lacework.net
```

To specify an AWS credential profile other than 'default':

```
lacework agent aws-install ec2ssm --credential_profile aws-profile-name
```

AWS credentials are read from the following environment variables: - AWS_ACCESS_KEY_ID - AWS_SECRET_ACCESS_KEY - AWS_SESSION_TOKEN (optional) - AWS_REGION

```
lacework agent aws-install ec2ssm [flags]
```

Options

```
      --credential_profile string   AWS credential profile to use (default "default")
  -d, --dry_run                     set this flag to print out the target instances and exit
  -f, --force_reinstall             set this flag to force-reinstall the agent, even if already
running on the target instance
  -h, --help                        help for ec2ssm
      --iam_role_name string        IAM role name (not ARN) with SSM policy, if not provided then
an ephemeral role will be created
  -r, --include_regions strings     list of regions to filter on
  -n, --max_parallelism int         maximum number of workers executing AWS API calls, set if rate
limits are lower or higher than normal (default 50)
      --server_url https://         server URL that agents will talk to, prefixed with https://
(default "https://agent.lacework.net")
      --skip_iam_role_creation      set this flag to skip creating an IAM role and instance
profile and associating the instance profile. Assumes all instances are already setup for SSM
      --tag strings                 only install agents on infra with this tag
      --tag_key string              only install agents on infra with this tag key set
      --token string                agent access token
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent aws-install - Install the datacollector agent on all remote AWS hosts

lacework agent gcp-install

Install the datacollector agent on all remote GCE hosts

Options

```
  -h, --help   help for gcp-install
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent - Manage Lacework agents
- lacework agent gcp-install osl - Use OSLogin to securely connect to GCE instances

lacework agent gcp-install osl

Use OSLogin to securely connect to GCE instances

Synopsis

This command installs the agent on all GCE instances in a GCP organization using OSLogin.

The username of the GCP user or service account, in the form `users/<username>`, is a required argument.

This command will attempt to query the GCE metadata server for the current project. If this command is not run on a GCE instance, pass the project ID as:

```
lacework agent gcp-install osl <gcp_username> --project_id my-project-id
```

To filter by one or more regions:

```
lacework agent gcp-install osl <gcp_username> --include_regions us-west1,europe-west2
```

To filter by instance metadata:

```
lacework agent gcp-install osl <gcp_username> --metadata MetadataKey,MetadataValue
```

To filter by instance metadata key:

```
lacework agent gcp-install osl <gcp_username> --metadata_key MetadataKey
```

To provide an agent access token of your choice, use the command 'lacework agent token list', select a token and pass it to the '--token' flag. This flag must be selected if the '--noninteractive' flag is set.

```
lacework agent gcp-install osl <gcp_username> --token <token>
```

To explicitly specify the server URL that the agent will connect to:

```
lacework agent gcp-install osl --server_url https://your.server.url.lacework.net
```

GCP credentials are read using the following environment variables: - GOOGLE_APPLICATION_CREDENTIALS

This command will automatically add hosts with successful connections to '~/.ssh/known_hosts' unless specified with '--trust_host_key=false'.

```
lacework agent gcp-install osl [flags]
```

Options

```
  -h, --help                     help for osl
  -r, --include_regions strings   list of regions to filter on
  -n, --max_parallelism int       maximum number of workers executing GCP API calls, set if rate
limits are lower or higher than normal (default 50)
      --metadata strings          only install agents on infra with this metadata
      --metadata_key string       only install agents on infra with this metadata key set
      --project_id string         ID of the GCP project, set if metadata server does not provide
      --server_url https://       server URL that agents will talk to, prefixed with https://
(default "https://agent.lacework.net")
      --token string              agent access token
      --trust_host_key            automatically add host keys to the ~/.ssh/known_hosts file
(default true)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent gcp-install - Install the datacollector agent on all remote GCE hosts

lacework agent install

Install the datacollector agent on a remote host

Synopsis

For single host installation of the Lacework agent via Secure Shell (SSH).

When this command is executed without any additional flag, an interactive prompt will be launched to help gather the necessary authentication information to access the remote host.

To authenticate to the remote host with a username and password.

```
lacework agent install <host> --ssh_username <your-user> --ssh_password <secret>
```

To authenticate to the remote host with an identity file instead.

```
lacework agent install <user@host> -i /path/to/your/key
```

To provide an agent access token of your choice, use the command 'lacework agent token list', select a token and pass it to the '--token' flag.

```
lacework agent install <user@host> -i /path/to/your/key --token <token>
```

To authenticate to the remote host on a non-standard SSH port use the '--ssh_port' flag or pass it directly via the argument.

```
lacework agent install <user@host:port>
```

To explicitly specify the server URL that the agent will connect to:

```
lacework agent install --server_url https://your.server.url.lacework.net
```

To list all active agents in your environment.

```
lacework agent list
```

NOTE: New agents could take up to an hour to report back to the platform.

```
lacework agent install <[user@]host[:port]> [flags]
```

Options

```
      --force                  override any pre-installed agent
  -h, --help                   help for install
  -i, --identity_file string   identity (private key) for public key authentication (default
"~/.ssh/id_rsa")
      --server_url https://    server URL that agents will talk to, prefixed with https://
(default "https://agent.lacework.net")
      --ssh_password string    password for authentication
      --ssh_port int           port to connect to on the remote host (default 22)
      --ssh_username string    username to login with
      --token string           agent access token
      --trust_host_key         automatically add host keys to the ~/.ssh/known_hosts file
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug                turn on debug logging
      --json                 switch commands output from human-readable to json format
      --nocache              turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework agent - Manage Lacework agents

lacework agent list

List all hosts with a running agent

Synopsis

List all hosts that have a running agent in your environment.

You can use 'key:value' pairs to filter the list of hosts with the --filter flag.

```
lacework agent list --filter 'os:Linux' --filter 'tags.VpcId:vpc-72225916'
```

NOTE: The value can be a regular expression such as 'hostname:db-server.*'

To filter hosts with a running agent version '5.8.0'.

```
lacework agent list --filter 'agentVersion:5.8.0.*' --filter 'status:ACTIVE'
```

The available keys for this command are: * agentVersion * hostname * ipAddr * mid * mode * os * status * tags.arch * tags.ExternalIp * tags.Hostname * tags.InstanceId * tags.InternalIp * tags.LwTokenShort * tags.os * tags.VmInstanceType * tags.VmProvider * tags.Zone * tags.Account * tags.AmiId * tags.Name * tags.SubnetId * tags.VpcId * tags.Cluster * tags.cluster-location * tags.cluster-name * tags.cluster-uid * tags.created-by * tags.enable-oslogin * tags.Env * tags.GCEtags * tags.gci-ensure-gke-docker * tags.gci-update-strategy * tags.google-compute-enable-pcid * tags.InstanceName * tags.InstanceTemplate * tags.kube-labels * tags.lw_KubernetesCluster * tags.NumericProjectId * tags.ProjectId

```
lacework agent list [flags]
```

Options

```
      --filter strings   filter results by key:value pairs (e.g. 'hostname:db-server.*')
  -h, --help             help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent - Manage Lacework agents

lacework agent token

Manage agent access tokens

Synopsis

Manage agent access tokens in your account.

Agent tokens should be treated as secret and not published. A token uniquely identifies a Lacework customer. If you suspect your token has been publicly exposed or compromised, generate a new token, update the new token on all machines using the old token. When complete, the old token can safely be disabled without interrupting Lacework services.

Options

```
  -h, --help   help for token
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent - Manage Lacework agents
- lacework agent token create - Create a new agent access token
- lacework agent token list - List all agent access tokens
- lacework agent token show - Show details about an agent access token
- lacework agent token update - Update an agent access token

lacework agent token create

Create a new agent access token

```
lacework agent token create <name> [description] [os] [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent token - Manage agent access tokens

lacework agent token list

List all agent access tokens

```
lacework agent token list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent token - Manage agent access tokens

lacework agent token show

Show details about an agent access token

```
lacework agent token show <token> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent token - Manage agent access tokens

lacework agent token update

Update an agent access token

Synopsis

Update an agent access token.

To update the token name and description:

```
lacework agent token update <token> --name dev --description "k8s deployment for dev"
```

To disable a token:

```
lacework agent token update <token> --disable
```

To enable a token:

```
lacework agent token update <token> --enable
```

```
lacework agent token update <token> [flags]
```

Options

```
      --description string   new agent access token description
      --disable              disable agent access token
      --enable               enable agent access token
  -h, --help                 help for update
      --name string          new agent access token name
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
```

```
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework agent token - Manage agent access tokens

lacework alert

Inspect and manage alerts

Synopsis

Inspect and manage alerts.

Lacework provides real-time alerts that are interactive and manageable.

Each alert contains various metadata information, such as severity level, type, status, alert category, and associated tags.

You can also post a comment to an alert's timeline; or change an alert status from Open to Closed.

For more information about alerts, visit:

https://docs.lacework.com/console/alerts-overview

To view all alerts in your Lacework account.

```
lacework alert ls
```

To show an alert.

```
lacework alert show <alert_id>
```

To close an alert.

```
lacework alert close <alert_id>
```

Options

```
  -h, --help   help for alert
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework alert close - Close an alert
- lacework alert comment - Add a comment
- lacework alert list - List all alerts
- lacework alert open - Open a specified alert in a web browser
- lacework alert show - Show details about a specific alert

lacework alert-channel

Manage alert channels

Synopsis

Manage alert channels integrations with Lacework

Options

```
  -h, --help   help for alert-channel
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework alert-channel create - Create a new alert channel integration
- lacework alert-channel delete - Delete a alert channel integration
- lacework alert-channel list - List all available alert channel integrations
- lacework alert-channel show - Show a single alert channel integration

lacework alert-channel create

Create a new alert channel integration

```
lacework alert-channel create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
```

```
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-channel - Manage alert channels

lacework alert-channel delete

Delete a alert channel integration

```
lacework alert-channel delete [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-channel - Manage alert channels

lacework alert-channel list

List all available alert channel integrations

```
lacework alert-channel list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-channel - Manage alert channels

lacework alert-channel show

Show a single alert channel integration

```
lacework alert-channel show [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-channel - Manage alert channels

lacework alert-profile

Manage alert profiles

Synopsis

Manage alert profiles to define how your LQL queries get consumed into alerts.

An alert profile consists of the ID of the new profile, the ID of an existing profile that the new profile extends, and a list of alert templates.

Options

```
  -h, --help   help for alert-profile
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework alert-profile create - Create a new alert profile
- lacework alert-profile delete - Delete an alert profile
- lacework alert-profile list - List all alert profiles
- lacework alert-profile show - Show an alert profile by ID
- lacework alert-profile update - Update alert templates from an existing alert profile

lacework alert-profile create

Create a new alert profile

```
lacework alert-profile create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-profile - Manage alert profiles

lacework alert-profile delete

Delete an alert profile

Synopsis

Delete a single alert profile by its ID.

```
lacework alert-profile delete <alert_profile_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-profile - Manage alert profiles

lacework alert-profile list

List all alert profiles

Synopsis

List all alert profiles configured in your Lacework account.

```
lacework alert-profile list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-profile - Manage alert profiles

lacework alert-profile show

Show an alert profile by ID

Synopsis

Show a single alert profile by its ID.

```
lacework alert-profile show <alert_profile_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-profile - Manage alert profiles

lacework alert-profile update

Update alert templates from an existing alert profile

```
lacework alert-profile update [alert_profile_id] [flags]
```

Options

```
  -h, --help   help for update
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-profile - Manage alert profiles

lacework alert-rule

Manage alert rules

Synopsis

Manage alert rules to route events to the appropriate people or tools.

An alert rule has three parts:

```
1. Alert channel(s) that should receive the event notification
2. Event severity and categories to include
3. Resource group(s) containing the subset of your environment to consider
```

Options

```
  -h, --help   help for alert-rule
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework alert-rule create - Create a new alert rule
- lacework alert-rule delete - Delete a alert rule
- lacework alert-rule list - List all alert rules
- lacework alert-rule show - Show an alert rule by ID

lacework alert-rule create

Create a new alert rule

```
lacework alert-rule create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-rule - Manage alert rules

lacework alert-rule delete

Delete a alert rule

Synopsis

Delete a single alert rule by it's ID.

```
lacework alert-rule delete <alert_rule_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-rule - Manage alert rules

lacework alert-rule list

List all alert rules

Synopsis

List all alert rules configured in your Lacework account.

```
lacework alert-rule list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-rule - Manage alert rules

lacework alert-rule show

Show an alert rule by ID

Synopsis

Show a single alert rule by it's ID.

```
lacework alert-rule show <alert_rule_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert-rule - Manage alert rules

lacework alert close

Close an alert

Synopsis

Use this command to change the status of an alert to closed.

The reason for closing the alert must be provided from the following options:

- 0 - Other
- 1 - False positive
- 2 - Not enough information
- 3 - Malicious and have resolution in place
- 4 - Expected because of routine testing.

Reasons may be provided inline or via prompt.

If you choose Other, a comment is required and should contain a brief explanation of why the alert is closed. Comments may be provided inline or via editor.

Note: A closed alert cannot be reopened. You will be prompted to confirm closure of the alert. This prompt can be bypassed with the --noninteractive flag

```
lacework alert close <alert_id> [flags]
```

Options

```
-c, --comment string   a comment to associate with the alert closure
-h, --help             help for close
-r, --reason int       the reason for closing the alert (default -1)
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert - Inspect and manage alerts

lacework alert comment

Add a comment

Synopsis

Post a user comment on an alert's timeline .

Comments may be provided inline or via editor.

```
lacework alert comment <alert_id> [flags]
```

Options

```
  -c, --comment string   a comment to add to the alert
  -h, --help             help for comment
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert - Inspect and manage alerts

lacework alert list

List all alerts

Synopsis

List all alerts.

By default, alerts are shown for the last 24 hours. Use a custom time range by suppling a range flag...

```
lacework alert ls --range "last 7 days"
```

Or by specifying start and end flags.

```
lacework alert ls --start "-7d@d" --end "now"
```

Start and end times may be specified in one of the following formats: A. A relative time specifier B. RFC3339 date and time C. Epoch time in milliseconds

To list open alerts of type "NewViolations" with high or critical severity.

```
lacework alert ls --status Open --severity high --type NewViolations
```

```
lacework alert list [flags]
```

Options

```
      --end string        end time for alerts (default "now")
  -h, --help              help for list
      --range string      natural time range for alerts
      --severity string   filter alerts by severity threshold (critical, high, medium, low, info)
      --start string      start time for alerts (default "-24h")
      --status string     filter alerts by status (Open, Closed)
      --type string       filter alerts by type
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert - Inspect and manage alerts

lacework alert open

Open a specified alert in a web browser

Synopsis

Open a specified alert in a web browser.

```
lacework alert open <alert_id> [flags]
```

Options

```
  -h, --help   help for open
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert - Inspect and manage alerts

lacework alert show

Show details about a specific alert

Synopsis

Show details about a specific alert.

There are different types of alert details that can be shown to assist with alert investigation. These types are referred to as alert detail scopes.

The following alert detail scopes are available:

- Details (default)
- Investigation
- Events
- RelatedAlerts
- Integrations
- Timeline
- ObservationTimeline

View an alert's timeline details:

```
lacework alert show <alert_id> --scope Timeline
```

```
lacework alert show <alert_id> [flags]
```

Options

```
  -h, --help           help for show
      --scope string   type of alert details to show (default "Details")
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework alert - Inspect and manage alerts

lacework api

Helper to call Lacework's API

Synopsis

Use this command as a helper to call any available Lacework API v2 endpoint.

API v2

To list all available Lacework schema types:

```
lacework api get /v2/schemas
```

To receive a json response of all machines within the given time window:

```
lacework api post /api/v2/Entities/Machines/search -d "{}"
```

To receive a json response of all agents within the given time window:

```
lacework api post /api/v2/AgentInfo/search -d "{}"
```

For a complete list of available API v2 endpoints visit:

```
https://<ACCOUNT>.lacework.net/api/v2/docs
```

```
lacework api <method> <path> [flags]
```

Options

```
  -d, --data string   data to send only for post and patch requests
  -h, --help          help for api
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache              turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.

lacework cloud-account

Manage cloud accounts

Synopsis

Manage cloud account integrations with Lacework

Options

```
  -h, --help   help for cloud-account
```

Options inherited from parent commands

```
  -a, --account string       account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string       access key id
  -s, --api_secret string    secret access key
      --api_token string     access token (replaces the use of api_key and api_secret)
      --debug                turn on debug logging
      --json                 switch commands output from human-readable to json format
      --nocache              turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework cloud-account create - Create a new cloud account integration
- lacework cloud-account delete - Delete a cloud account integration
- lacework cloud-account list - List all available cloud account integrations
- lacework cloud-account migrate - Mark a GCPv1 (storage-based) cloud account integration for migration
- lacework cloud-account show - Show a single cloud account integration

lacework cloud-account create

Create a new cloud account integration

```
lacework cloud-account create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework cloud-account - Manage cloud accounts

lacework cloud-account delete

Delete a cloud account integration

```
lacework cloud-account delete [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework cloud-account - Manage cloud accounts

lacework cloud-account list

List all available cloud account integrations

```
lacework cloud-account list [flags]
```

Options

```
  -h, --help          help for list
  -t, --type string   list all cloud accounts of a specific type
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework cloud-account - Manage cloud accounts

lacework cloud-account migrate

Mark a GCPv1 (storage-based) cloud account integration for migration

```
lacework cloud-account migrate [flags]
```

Options

```
  -h, --help   help for migrate
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework cloud-account - Manage cloud accounts

lacework cloud-account show

Show a single cloud account integration

```
lacework cloud-account show [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework cloud-account - Manage cloud accounts

lacework completion

Generate the autocompletion script for the specified shell

Synopsis

Generate the autocompletion script for lacework for the specified shell. See each sub-command's help for details on how to use the generated script.

Options

```
  -h, --help   help for completion
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework completion bash - Generate the autocompletion script for bash
- lacework completion fish - Generate the autocompletion script for fish
- lacework completion powershell - Generate the autocompletion script for powershell
- lacework completion zsh - Generate the autocompletion script for zsh

lacework completion bash

Generate the autocompletion script for bash

Synopsis

Generate the autocompletion script for the bash shell.

This script depends on the 'bash-completion' package. If it is not installed already, you can install it via your OS's package manager.

To load completions in your current shell session:

```
source <(lacework completion bash)
```

To load completions for every new session, execute once:

Linux:

```
lacework completion bash > /etc/bash_completion.d/lacework
```

macOS:

```
lacework completion bash > $(brew --prefix)/etc/bash_completion.d/lacework
```

You will need to start a new shell for this setup to take effect.

```
lacework completion bash
```

Options

```
  -h, --help              help for bash
      --no-descriptions   disable completion descriptions
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework completion - Generate the autocompletion script for the specified shell

lacework completion fish

Generate the autocompletion script for fish

Synopsis

Generate the autocompletion script for the fish shell.

To load completions in your current shell session:

```
lacework completion fish | source
```

To load completions for every new session, execute once:

```
lacework completion fish > ~/.config/fish/completions/lacework.fish
```

You will need to start a new shell for this setup to take effect.

```
lacework completion fish [flags]
```

Options

```
  -h, --help              help for fish
      --no-descriptions   disable completion descriptions
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework completion - Generate the autocompletion script for the specified shell

lacework completion powershell

Generate the autocompletion script for powershell

Synopsis

Generate the autocompletion script for powershell.

To load completions in your current shell session:

```
lacework completion powershell | Out-String | Invoke-Expression
```

To load completions for every new session, add the output of the above command to your powershell profile.

```
lacework completion powershell [flags]
```

Options

```
  -h, --help              help for powershell
      --no-descriptions   disable completion descriptions
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework completion - Generate the autocompletion script for the specified shell

lacework completion zsh

Generate the autocompletion script for zsh

Synopsis

Generate the autocompletion script for the zsh shell.

If shell completion is not already enabled in your environment you will need to enable it. You can execute the following once:

```
echo "autoload -U compinit; compinit" >> ~/.zshrc
```

To load completions in your current shell session:

```
source <(lacework completion zsh)
```

To load completions for every new session, execute once:

Linux:

```
lacework completion zsh > "${fpath[1]}/_lacework"
```

macOS:

```
lacework completion zsh > $(brew --prefix)/share/zsh/site-functions/_lacework
```

You will need to start a new shell for this setup to take effect.

```
lacework completion zsh [flags]
```

Options

```
  -h, --help              help for zsh
      --no-descriptions   disable completion descriptions
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework completion - Generate the autocompletion script for the specified shell

lacework compliance

Manage compliance reports

Synopsis

Manage compliance reports for Google, Azure, or AWS cloud providers.

Lacework cloud security platform provides continuous Compliance monitoring against cloud security best practices and compliance standards as CIS, PCI DSS, SoC II and HIPAA benchmark standards.

Get started by integrating one or more cloud accounts using the command:

```
lacework cloud-account create
```

If you prefer to configure the integration via the WebUI, log in to your account at:

```
https://<ACCOUNT>.lacework.net
```

Then navigate to Settings > Integrations > Cloud Accounts.

Use the following command to list all available integrations in your account:

```
lacework cloud-account list
```

Options

```
  -h, --help   help for compliance
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework compliance aws - Compliance for AWS
- lacework compliance azure - Compliance for Azure Cloud
- lacework compliance google - Compliance for Google Cloud

lacework compliance aws

Compliance for AWS

Synopsis

Manage compliance reports for Amazon Web Services (AWS).

To list all AWS accounts configured in your account:

```
lacework compliance aws list-accounts
```

To get the latest AWS compliance assessment report:

```
lacework compliance aws get-report <account_id>
```

These reports run on a regular schedule, typically once a day.

Options

```
  -h, --help   help for aws
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance - Manage compliance reports
- lacework compliance aws get-report - Get the latest AWS compliance report
- lacework compliance aws list-accounts - List all AWS accounts configured
- lacework compliance aws scan - Scan triggers a new resource inventory scan
- lacework compliance aws search - Search for all known violations of a given resource arn

lacework compliance aws get-report

Get the latest AWS compliance report

Synopsis

Get the latest compliance assessment report from the provided AWS account, these reports run on a regular schedule, typically once a day. The available report formats are human-readable (default), json and pdf.

To list all AWS accounts configured in your account:

```
lacework compliance aws list-accounts
```

To show recommendation details and affected resources for a recommendation id:

```
lacework compliance aws get-report <account_id> [recommendation_id]
```

To retrieve a specific report by its report name:

```
lacework compliance aws get-report <account_id> --report_name 'AWS CSA CCM 4.0.5'
```

```
lacework compliance aws get-report <account_id> [recommendation_id] [flags]
```

Options

```
      --category strings     filter report details by category (identity-and-access-management,
s3, logging...)
      --csv                  output report in CSV format
      --details              increase details about the compliance report
```

```
  -h, --help                 help for get-report
      --pdf                  download report in PDF format
      --report_name string   report name to display, run 'lacework report-definitions list' for
more information. (default "CIS Amazon Web Services Foundations Benchmark v1.4.0")
      --service strings      filter report details by service (aws:s3, aws:iam, aws:cloudtrail,
...)
      --severity string      filter report details by severity threshold (critical, high, medium,
low, info)
      --status string        filter report details by status (non-compliant, requires-manual-
assessment, suppressed, compliant, could-not-assess)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance aws - Compliance for AWS

lacework compliance aws list-accounts

List all AWS accounts configured

Synopsis

List all AWS accounts configured in your account.

```
lacework compliance aws list-accounts [flags]
```

Options

```
  -h, --help   help for list-accounts
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance aws - Compliance for AWS

lacework compliance aws scan

Scan triggers a new resource inventory scan

Synopsis

Scan triggers a new resource inventory scan.

```
lacework compliance aws scan [flags]
```

Options

```
  -h, --help   help for scan
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance aws - Compliance for AWS

lacework compliance aws search

Search for all known violations of a given resource arn

Synopsis

Search for all known violations of a given resource arn.

```
lacework compliance aws search <resource_arn> [flags]
```

Options

```
  -h, --help   help for search
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance aws - Compliance for AWS

lacework compliance azure

Compliance for Azure Cloud

Synopsis

Manage compliance reports for Azure Cloud.

To list all Azure tenants configured in your account:

```
lacework compliance azure list-tenants
```

To list all Azure subscriptions from a tenant, use the command:

```
lacework compliance azure list-subscriptions <tenant_id>
```

To get the latest Azure compliance assessment report, use the command:

```
lacework compliance azure get-report <tenant_id> <subscription_id>
```

These reports run on a regular schedule, typically once a day.

Options

```
  -h, --help   help for azure
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance - Manage compliance reports
- lacework compliance azure get-report - Get the latest Azure compliance report
- lacework compliance azure list - List Azure tenants and subscriptions
- lacework compliance azure list-subscriptions - List subscriptions `<tenant-id>`
- lacework compliance azure scan - Scan triggers a new resource inventory scan

lacework compliance azure get-report

Get the latest Azure compliance report

Synopsis

Get the latest Azure compliance assessment report, these reports run on a regular schedule, typically once a day. The available report formats are human-readable (default), json and pdf.

To list all Azure tenants and subscriptions configured in your account:

```
lacework compliance azure list
```

To show recommendation details and affected resources for a recommendation id:

```
lacework compliance azure get-report <tenant_id> <subscriptions_id> [recommendation_id]
```

To retrieve a specific report by its report name:

```
lacework compliance azure get-report <tenant_id> <subscriptions_id> --report_name 'Azure CIS 1.3.1
Report'
```

```
lacework compliance azure get-report <tenant_id> <subscriptions_id> [flags]
```

Options

```
      --category strings     filter report details by category (networking, storage, ...)
      --csv                  output report in CSV format
      --details              increase details about the compliance report
  -h, --help                 help for get-report
      --pdf                  download report in PDF format
      --report_name string   report name to display, run 'lacework report-definitions list' for
more information. (default "CIS Microsoft Azure Foundations Benchmark v1.5.0")
      --service strings      filter report details by service (azure:ms:storage, azure:ms:sql,
azure:ms:network, ...)
      --severity string      filter report details by severity threshold (critical, high, medium,
low, info)
      --status string        filter report details by status (non-compliant, requires-manual-
assessment, suppressed, compliant, could-not-assess)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance azure - Compliance for Azure Cloud

lacework compliance azure list

List Azure tenants and subscriptions

Synopsis

List all Azure tenants and subscriptions configured in your account.

```
lacework compliance azure list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance azure - Compliance for Azure Cloud

lacework compliance azure list-subscriptions

List subscriptions `<tenant-id>`

Synopsis

List all Azure subscriptions for Tenant.

Use the following command to list all Azure Tenants configured in your account:

```
lacework compliance az list
```

```
lacework compliance azure list-subscriptions [flags]
```

Options

```
  -h, --help   help for list-subscriptions
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance azure - Compliance for Azure Cloud

lacework compliance azure scan

Scan triggers a new resource inventory scan

Synopsis

Scan triggers a new resource inventory scan.

```
lacework compliance azure scan [flags]
```

Options

```
  -h, --help   help for scan
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance azure - Compliance for Azure Cloud

lacework compliance google

Compliance for Google Cloud

Synopsis

Manage compliance reports for Google Cloud.

To list all GCP organizations and projects configured in your account:

```
lacework compliance gcp list
```

To list all GCP projects from an organization, use the command:

```
lacework compliance gcp list-projects <organization_id>
```

To get the latest GCP compliance assessment report, use the command:

```
lacework compliance gcp get-report <organization_id> <project_id>
```

These reports run on a regular schedule, typically once a day.

Options

```
  -h, --help   help for google
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance - Manage compliance reports
- lacework compliance google get-report - Get the latest GCP compliance report
- lacework compliance google list - List gcp projects and organizations
- lacework compliance google list-projects - List projects from an organization
- lacework compliance google scan - Scan triggers a new resource inventory scan

lacework compliance google get-report

Get the latest GCP compliance report

Synopsis

Get the latest compliance assessment report, these reports run on a regular schedule, typically once a day. The available report formats are human-readable (default), json and pdf.

To list all GCP projects and organizations configured in your account:

```
lacework compliance gcp list
```

To show recommendation details and affected resources for a recommendation id:

```
lacework compliance gcp get-report <organization_id> <project_id> [recommendation_id]
```

To retrieve a specific report by its report name:

```
lacework compliance gcp get-report <organization_id> <project_id> --report_name 'GCP Cybersecurity
Maturity'
```

```
lacework compliance google get-report <organization_id> <project_id> [flags]
```

Options

```
      --category strings     filter report details by category (storage, networking, identity-and-
access-management, ...)
      --csv                  output report in CSV format
      --details              increase details about the compliance report
  -h, --help                 help for get-report
      --pdf                  download report in PDF format
      --report_name string   report name to display, run 'lacework report-definitions list' for
more information. (default "GCP CIS Benchmark 1.3")
      --service strings      filter report details by service (gcp:storage:bucket,
gcp:kms:cryptoKey, gcp:project, ...)
      --severity string      filter report details by severity threshold (critical, high, medium,
low, info)
      --status string        filter report details by status (non-compliant, requires-manual-
assessment, suppressed, compliant, could-not-assess)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance google - Compliance for Google Cloud

lacework compliance google list

List gcp projects and organizations

Synopsis

List all GCP projects and organization IDs.

```
lacework compliance google list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance google - Compliance for Google Cloud

lacework compliance google list-projects

List projects from an organization

Synopsis

List all GCP projects from the provided organization ID.

Use the following command to list all GCP integrations in your account:

```
lacework cloud-account list --type GcpCfg
```

Then, select one GUID from an integration and visualize its details using the command:

```
lacework cloud-account show <int_guid>
```

```
lacework compliance google list-projects <organization_id> [flags]
```

Options

```
  -h, --help   help for list-projects
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance google - Compliance for Google Cloud

lacework compliance google scan

Scan triggers a new resource inventory scan

Synopsis

Scan triggers a new resource inventory scan.

```
lacework compliance google scan [flags]
```

Options

```
  -h, --help   help for scan
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
```

```
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework compliance google - Compliance for Google Cloud

lacework component

Manage components

Synopsis

Manage components to extend your experience with the Lacework platform

Options

```
-h, --help   help for component
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
```

```
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework component install - Install a new component
- lacework component list - List all components
- lacework component show - Show details about a component
- lacework component uninstall - Uninstall an existing component
- lacework component update - Update an existing component

lacework component install

Install a new component

Synopsis

Install a new component

```
lacework component install <component> [flags]
```

Options

```
  -h, --help             help for install
      --version string   require a specific version to be installed (default is latest)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework component - Manage components

lacework component list

List all components

Synopsis

List all available components and their current state

```
lacework component list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework component - Manage components

lacework component show

Show details about a component

Synopsis

Show details about a component

```
lacework component show <component> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework component - Manage components

lacework component uninstall

Uninstall an existing component

Synopsis

Uninstall an existing component

```
lacework component uninstall <component> [flags]
```

Options

```
  -h, --help   help for uninstall
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework component - Manage components

lacework component update

Update an existing component

Synopsis

Update an existing component

```
lacework component update <component> [flags]
```

Options

```
  -h, --help             help for update
      --version string   update to a specific version (default is latest)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework component - Manage components

lacework configure

Configure the Lacework CLI

Synopsis

Configure settings that the Lacework CLI uses to interact with the Lacework platform. These include your Lacework account, API access key and secret.

To create a set of API keys, log in to your Lacework account via WebUI and navigate to Settings > API Keys and click + Create New. Enter a name for the key and an optional description, then click Save. To get the secret key, download the generated API key file.

Use the flag --json_file to preload the downloaded API key file. Use the flag --txt_file to preload the downloaded API key file from the FortiCloud portal.

If this command is run with no flags, the Lacework CLI will store all settings under the default profile. The information in the default profile is used any time you run a Lacework CLI command that doesn't explicitly specify a profile to use.

You can configure multiple profiles by using the --profile flag. If a config file does not exist (the default location is ~/.lacework.toml), the Lacework CLI will create it for you.

```
lacework configure [flags]
```

Options

```
  -h, --help               help for configure
  -j, --json_file string   loads the API key JSON file downloaded from the WebUI
  -t, --txt_file string    loads the API key TXT file downloaded from the FortiCloud portal
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework configure list - List all configured profiles at ~/.lacework.toml
- lacework configure show - Show current configuration data
- lacework configure switch-profile - Switch between configured profiles

lacework configure list

List all configured profiles at ~/.lacework.toml

Synopsis

List all profiles configured into the config file ~/.lacework.toml

To switch profiles permanently use the command.

```
lacework configure switch-profile profile2
```

```
lacework configure list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework configure - Configure the Lacework CLI

lacework configure show

Show current configuration data

Synopsis

Prints the current computed configuration data from the specified configuration key. The order of precedence to compute the configuration is flags, environment variables, and the configuration file ~/.lacework.toml.

The available configuration keys are:

- profile
- account
- subaccount
- api_secret
- api_key

To show the configuration from a different profile, use the flag --profile.

```
lacework configure show account --profile my-profile
```

```
lacework configure show <config_key> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework configure - Configure the Lacework CLI

lacework configure switch-profile

Switch between configured profiles

Synopsis

Switch between profiles configured into the config file ~/.lacework.toml

An alternative to temporarily switch to a different profile in your current terminal is to export the environment variable:

```
$env:LW_PROFILE = 'my-profile'
```

```
lacework configure switch-profile <profile> [flags]
```

Options

```
  -h, --help   help for switch-profile
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework configure - Configure the Lacework CLI

lacework container-registry

Manage container registries

Synopsis

Manage container registry integrations with Lacework

Options

```
  -h, --help   help for container-registry
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework container-registry create - Create a new container registry integration
- lacework container-registry delete - Delete a container registry integration
- lacework container-registry list - List all available container registry integrations
- lacework container-registry show - Show a single container registry integration

lacework container-registry create

Create a new container registry integration

```
lacework container-registry create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework container-registry - Manage container registries

lacework container-registry delete

Delete a container registry integration

```
lacework container-registry delete [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework container-registry - Manage container registries

lacework container-registry list

List all available container registry integrations

```
lacework container-registry list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework container-registry - Manage container registries

lacework container-registry show

Show a single container registry integration

```
lacework container-registry show [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework container-registry - Manage container registries

lacework generate

Generate code to onboard your account

Synopsis

Generate code to onboard your account and deploy Lacework into various cloud environments.

This command creates Infrastructure as Code (IaC) in the form of Terraform HCL, with the option of running Terraform and deploying Lacework into AWS, Azure, GCP or OCI.

Options

```
      --apply           run terraform apply without executing plan or prompting
  -h, --help            help for generate
      --output string   location to write generated content
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework generate cloud-account - Generate cloud integration IaC
- lacework generate k8s - Generate Kubernetes integration IaC

lacework generate cloud-account

Generate cloud integration IaC

Synopsis

Generate cloud-account IaC to deploy Lacework into a cloud environment.

This command creates Infrastructure as Code (IaC) in the form of Terraform HCL, with the option of running Terraform and deploying Lacework into AWS, Azure, GCP or OCI.

Options

```
  -h, --help   help for cloud-account
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --apply               run terraform apply without executing plan or prompting
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
      --output string       location to write generated content
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate - Generate code to onboard your account
- lacework generate cloud-account aws - Generate and/or execute Terraform code for AWS integration
- lacework generate cloud-account azure - Generate and/or execute Terraform code for Azure integration
- lacework generate cloud-account gcp - Generate and/or execute Terraform code for GCP integration
- lacework generate cloud-account oci - Generate and/or execute Terraform code for OCI integration

lacework generate cloud-account aws

Generate and/or execute Terraform code for AWS integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into an AWS environment.

By default, this command interactively prompts for the required information to setup the new cloud account. In interactive mode, this command will:

- Prompt for the required information to setup the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
  - Once Terraform is detected or installed, Terraform plan will be executed
- The command will prompt with the outcome of the plan and allow to view more details or continue with Terraform apply
  - If confirmed, Terraform apply will be run, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter value (s) required for Terraform code generation.

```
lacework generate cloud-account aws [flags]
```

Options

```
      --agentless                                 enable agentless integration
      --agentless_management_account_id string    AWS management account ID for Agentless
integration
      --agentless_monitored_account_ids strings   AWS monitored account IDs for Agentless
```

```
integrations; may contain account IDs, OUs, or the organization root (e.g. 123456789000,ou-abcd-
12345678,r-abcd)
      --agentless_monitored_accounts strings      AWS monitored accounts for Agentless
integrations; value format must be <aws profile>:<region>
      --agentless_scanning_accounts strings       AWS scanning accounts for Agentless
integrations; value format must be <aws profile>:<region>
      --apply                                     run terraform apply without executing plan or
prompting
      --aws_assume_role string                    specify aws assume role
      --aws_organization                          enable organization integration
      --aws_profile string                        specify aws profile
      --aws_region string                         specify aws region
      --aws_subaccount strings                    configure an additional aws account; value
format must be <aws profile>:<region>
      --bucket_encryption_enabled                 enable S3 bucket encryption when creating bucket
(default true)
      --bucket_name string                        specify bucket name when creating bucket
      --bucket_sse_key_arn string                 specify existing KMS encryption key arn for
bucket
      --cloudtrail                                enable cloudtrail integration
      --cloudtrail_name string                    specify name of cloudtrail integration
      --cloudtrail_org_account_mapping string     Org account mapping json string. Example: '
{"default_lacework_account":"main", "mapping": [{ "aws_accounts": ["123456789011"], "lacework_
account": "sub-account-1"}]}'
      --config                                    enable config integration
      --config_cf_resource_prefix string          specify Cloudformation resource prefix for
Config organization integration
      --config_lacework_access_key_id string      specify AWS access key ID for Config
organization integration
      --config_lacework_account string            specify lacework account for Config organization
integration
      --config_lacework_secret_key string         specify AWS secret key for Config organization
integration
      --config_lacework_sub_account string        specify lacework sub-account for Config
organization integration
      --config_organization_id string             specify AWS organization ID for Config
organization integration
      --config_organization_units strings         specify AWS organization units for Config
organization integration
      --consolidated_cloudtrail                   use consolidated trail
      --controltower                              enable Control Tower integration
      --controltower_audit_account string         specify AWS Control Tower Audit account; value
format must be <aws profile>:<region>
      --controltower_kms_key_arn string           specify AWS Control Tower custom kMS key ARN
      --controltower_log_archive_account string   specify AWS Control Tower Log Archive account;
value format must be <aws profile>:<region>
      --existing_bucket_arn string                specify existing cloudtrail S3 bucket ARN
      --existing_iam_role_arn string              specify existing iam role arn to use
      --existing_iam_role_externalid string       specify existing iam role external_id to use
      --existing_iam_role_name string             specify existing iam role name to use
      --existing_sns_topic_arn string             specify existing SNS topic arn
  -h, --help                                      help for aws
```

```
      --lacework_aws_account_id string            the Lacework AWS root account id
      --output string                             location to write generated content (default is
~/lacework/aws)
      --sns_topic_encryption_enabled              enable encryption on SNS topic when creating one
(default true)
      --sns_topic_encryption_key_arn string       specify existing KMS encryption key arn for SNS
topic
      --sns_topic_name string                     specify SNS topic name if creating new one
      --sqs_encryption_enabled                    enable encryption on SQS queue when creating
(default true)
      --sqs_encryption_key_arn string             specify existing KMS encryption key arn for SQS
queue
      --sqs_queue_name string                     specify SQS queue name if creating new one
      --use_s3_bucket_notification                enable S3 bucket notifications
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate cloud-account - Generate cloud integration IaC
- lacework generate cloud-account aws controltower - Generate and/or execute Terraform code for ControlTower integration

lacework generate cloud-account aws controltower

Generate and/or execute Terraform code for ControlTower integration

Synopsis

Use this command to generate Terraform code for deploying Lacework with Aws Cloudtrail and ControlTower.

By default, this command interactively prompts for the required information to set up the new cloud account. In interactive mode, this command will:

- Prompt for the required information to set up the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
- Once Terraform is detected or installed, the Terraform plan is executed
- The command prompts you with the outcome of the plan and allows you to view more details or continue with Terraform apply
- If confirmed, Terraform apply runs, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter values required for Terraform code generation.

```
lacework generate cloud-account aws controltower [flags]
```

Options

```
      --apply                          run terraform apply without executing plan or prompting
      --audit_account string           The audit account flag input in the format profile:region
  -h, --help                           help for controltower
      --iam_role_arn string            specify the arn of the existing iam role
      --iam_role_external_id string    specify the external id of the existing iam role
      --iam_role_name string           specify the name of the existing iam role
      --lacework_aws_account_id string   the Lacework AWS root account id
      --log_archive_account string     The log archive account flag input in the format
profile:region
      --org_account_mapping string     Org account mapping json string. Example: '{"default_
lacework_account":"main", "mapping": [{ "aws_accounts": ["123456789011"], "lacework_account":
"sub-account-1"}]}'
      --output string                  location to write generated content
      --prefix string                  specify the prefix that will be used at the beginning of
every generated resource
      --s3_bucket_arn string           the S3 Bucket for consolidated CloudTrail
      --sns_topic_arn string           the SNS Topic
      --sqs_queue_name string          specify the name of the sqs queue
```

Options inherited from parent commands

```
  -a, --account string                            account subdomain of URL (i.e.
<ACCOUNT>.lacework.net)
      --agentless                                 enable agentless integration
      --agentless_management_account_id string    AWS management account ID for Agentless
integration
      --agentless_monitored_account_ids strings   AWS monitored account IDs for Agentless
integrations; may contain account IDs, OUs, or the organization root (e.g. 123456789000,ou-abcd-
12345678,r-abcd)
      --agentless_monitored_accounts strings      AWS monitored accounts for Agentless
integrations; value format must be <aws profile>:<region>
      --agentless_scanning_accounts strings       AWS scanning accounts for Agentless
integrations; value format must be <aws profile>:<region>
  -k, --api_key string                            access key id
  -s, --api_secret string                         secret access key
      --api_token string                          access token (replaces the use of api_key and
api_secret)
      --aws_assume_role string                    specify aws assume role
      --aws_organization                          enable organization integration
      --aws_profile string                        specify aws profile
      --aws_region string                         specify aws region
      --aws_subaccount strings                    configure an additional aws account; value
format must be <aws profile>:<region>
      --bucket_encryption_enabled                 enable S3 bucket encryption when creating bucket
(default true)
      --bucket_name string                        specify bucket name when creating bucket
      --bucket_sse_key_arn string                 specify existing KMS encryption key arn for
bucket
      --cloudtrail                                enable cloudtrail integration
      --cloudtrail_name string                    specify name of cloudtrail integration
      --cloudtrail_org_account_mapping string     Org account mapping json string. Example: '
{"default_lacework_account":"main", "mapping": [{ "aws_accounts": ["123456789011"], "lacework_
account": "sub-account-1"}]}'
      --config                                    enable config integration
      --config_cf_resource_prefix string          specify Cloudformation resource prefix for
Config organization integration
      --config_lacework_access_key_id string      specify AWS access key ID for Config
organization integration
      --config_lacework_account string            specify lacework account for Config organization
integration
      --config_lacework_secret_key string         specify AWS secret key for Config organization
integration
      --config_lacework_sub_account string        specify lacework sub-account for Config
organization integration
      --config_organization_id string             specify AWS organization ID for Config
organization integration
      --config_organization_units strings         specify AWS organization units for Config
organization integration
      --consolidated_cloudtrail                   use consolidated trail
      --controltower                              enable Control Tower integration
      --controltower_audit_account string         specify AWS Control Tower Audit account; value
format must be <aws profile>:<region>
      --controltower_kms_key_arn string           specify AWS Control Tower custom kMS key ARN
      --controltower_log_archive_account string   specify AWS Control Tower Log Archive account;
value format must be <aws profile>:<region>
      --debug                                     turn on debug logging
      --existing_bucket_arn string                specify existing cloudtrail S3 bucket ARN
      --existing_iam_role_arn string              specify existing iam role arn to use
      --existing_iam_role_externalid string       specify existing iam role external_id to use
      --existing_iam_role_name string             specify existing iam role name to use
      --existing_sns_topic_arn string             specify existing SNS topic arn
      --json                                      switch commands output from human-readable to
json format
      --nocache                                   turn off caching
      --nocolor                                   turn off colors
      --noninteractive                            turn off interactive mode (disable spinners,
prompts, etc.)
      --organization                              access organization level data sets (org admins
only)
  -p, --profile string                            switch between profiles configured at
~/.lacework.toml
      --sns_topic_encryption_enabled              enable encryption on SNS topic when creating one
(default true)
      --sns_topic_encryption_key_arn string       specify existing KMS encryption key arn for SNS
topic
      --sns_topic_name string                     specify SNS topic name if creating new one
      --sqs_encryption_enabled                    enable encryption on SQS queue when creating
(default true)
      --sqs_encryption_key_arn string             specify existing KMS encryption key arn for SQS
queue
      --subaccount string                         sub-account name inside your organization (org
admins only)
      --use_s3_bucket_notification                enable S3 bucket notifications
```

See also

- lacework generate cloud-account aws - Generate and/or execute Terraform code for AWS integration

lacework generate cloud-account azure

Generate and/or execute Terraform code for Azure integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into new Azure environment.

By default, this command will function interactively, prompting for the required information to setup the new cloud account. In interactive mode, this command will:

- Prompt for the required information to setup the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version will be confirmed suitable for use
- If Terraform is not installed, or the version installed is not suitable, a new version will be installed into a temporary location
- Once Terraform is detected or installed, Terraform plan will be executed
- The command will prompt with the outcome of the plan and allow to view more details or continue with Terraform apply
- If confirmed, Terraform apply will be run, completing the setup of the cloud account

```
lacework generate cloud-account azure [flags]
```

Options

```
      --activity_log                                    enable activity log integration
      --activity_log_integration_name string            specify a custom activity log integration
name
      --ad_create                                       create new active directory integration
(default true)
      --ad_id string                                    existing active directory application id
      --ad_pass string                                  existing active directory application
password
      --ad_pid string                                   existing active directory application
service principle id
      --agentless                                       enable agentless integration
      --agentless_subscription_ids strings              comma-separated list of subscription IDs
for Agentless scanning (e.g., 'sub1,sub2,sub3')
      --all_subscriptions subscription ids              grant read access to ALL subscriptions
within Tenant (overrides subscription ids)
      --apply                                           run terraform apply for the generated hcl
      --configuration                                   enable configuration integration
      --configuration_name string                       specify a custom configuration integration
name
      --create_log_analytics_workspace                  enable creation of Log Analytics Workspace
for agentless scanning
      --entra_id_activity_log                           enable Entra ID activity log integration
      --entra_id_activity_log_integration_name string   specify a custom Entra ID activity log
integration name
      --event_hub_location string                       specify the location where the Event Hub
for logging will reside
      --event_hub_partition_count int                   specify the number of partitions for the
Event Hub (default 1)
      --existing_storage                                use existing storage account
      --global                                          enable global agentless scanning
```

```
  -h, --help                                            help for azure
      --integration_level string                        specify the agentless integration level
(e.g., 'SUBSCRIPTION', 'TENANT')
      --location string                                 specify azure region where storage account
logging resides
      --management_group                                management group level integration
      --management_group_id string                      specify management group id. Required if
mgmt_group provided
      --output string                                   location to write generated content
(default is ~/lacework/azure)
      --regions strings                                 comma-separated list of Azure regions for
agentless scanning (e.g., 'East US,West US')
      --storage_account_name string                     specify storage account name
      --storage_account_network_rule_ip_rules strings   list of IP rules to apply to the storage
account network rules; format is ip1,ip2,ip3
      --storage_resource_group string                   specify storage resource group
      --subscription_id string                          specify the Azure Subscription ID to be
used to provision Lacework resources
      --subscription_ids strings                        list of subscriptions to grant read
access; format is id1,id2,id3
      --use_storage_account_network_rules               enable storage account network rules
(default true)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate cloud-account - Generate cloud integration IaC

lacework generate cloud-account gcp

Generate and/or execute Terraform code for GCP integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into an GCP environment.

By default, this command interactively prompts for the required information to setup the new cloud account. In interactive mode, this command will:

- Prompt for the required information to setup the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
- Once Terraform is detected or installed, Terraform plan will be executed
- The command will prompt with the outcome of the plan and allow to view more details or continue with Terraform apply
- If confirmed, Terraform apply will be run, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter value (s) required for Terraform code generation.

```
lacework generate cloud-account gcp [flags]
```

Options

```
      --agentless                                    enable agentless integration
      --apply                                        run terraform apply without executing plan
or prompting
      --audit_log                                    enable audit log integration
      --audit_log_integration_name string            specify a custom audit log integration name
      --configuration                                enable configuration integration
      --configuration_integration_name string        specify a custom configuration integration
name
      --custom_filter string                         Audit Log filter which supersedes all other
filter options when defined
      --existing_service_account_name string         specify existing service account name
      --existing_service_account_private_key string  specify existing service account private key
(base64 encoded)
      --existing_sink_name string                    specify existing sink name
  -e, --folders_to_exclude stringArray               List of root folders to exclude for an
organization-level integration
```

```
  -i, --folders_to_include stringArray                 list of root folders to include for an 
organization-level integration
      --google_workspace_filter                        filter out Google Workspace login logs from 
GCP Audit Log sinks (default true)
  -h, --help                                           help for gcp
      --include_root_projects                          Disables logic that includes root-level 
projects if excluding folders (default true)
      --k8s_filter                                     filter out GKE logs from GCP Audit Log sinks 
(default true)
      --organization_id string                         specify the organization id (only set if 
agentless integration or organization_integration is set)
      --organization_integration                       enable organization integration
      --output string                                  location to write generated content (default 
is ~/lacework/gcp)
      --prefix string                                  prefix that will be used at the beginning of 
every generated resource
      --project_filter_list strings                    List of GCP project IDs to monitor for 
Agentless integration
      --project_id string                              specify the project id to be used to 
provision lacework resources (required)
      --projects strings                               list of project IDs to integrate with 
(project-level integrations)
      --regions strings                                List of GCP regions to deploy for Agentless 
integration
      --service_account_credentials string             specify service account credentials JSON 
file path (leave blank to make use of google credential ENV vars)
      --use_pub_sub                                    deprecated: pub/sub audit log integration is 
always used and only supported type (default true)
      --wait_time string                               amount of time to wait before the next 
resource is provisioned
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate cloud-account - Generate cloud integration IaC

lacework generate cloud-account oci

Generate and/or execute Terraform code for OCI integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into an OCI tenant.

By default, this command interactively prompts for the required information to setup the new cloud account. In interactive mode, this command will:

- Prompt for the required information to setup the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
  - Once Terraform is detected or installed, Terraform plan will be executed
- The command will prompt with the outcome of the plan and allow to view more details or continue with Terraform apply
  - If confirmed, Terraform apply will be run, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter value (s) required for Terraform code generation.

```
lacework generate cloud-account oci [flags]
```

Options

```
      --apply                    run terraform apply without executing plan or prompting
      --config                   enable configuration integration
      --config_name string       specify name of configuration integration
  -h, --help                     help for oci
      --oci_user_email string    specify the email address to associate with the integration OCI
user
      --output string            location to write generated content (default is ~/lacework/oci)
      --tenant_ocid string       specify the OCID of the tenant to integrate
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate cloud-account - Generate cloud integration IaC

lacework generate k8s

Generate Kubernetes integration IaC

Synopsis

Generate IaC to deploy Lacework into a Kubernetes platform.

This command creates Infrastructure as Code (IaC) in the form of Terraform HCL, with the option of running Terraform and deploying Lacework into GKE.

Options

```
  -h, --help   help for k8s
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --apply               run terraform apply without executing plan or prompting
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
      --output string       location to write generated content
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate - Generate code to onboard your account
- lacework generate k8s eks - Generate and/or execute Terraform code for EKS integration
- lacework generate k8s gke - Generate and/or execute Terraform code for GKE integration

lacework generate k8s eks

Generate and/or execute Terraform code for EKS integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into an EKS environment.

By default, this command interactively prompts for the required information to set up the new cloud account. In interactive mode, this command will:

- Prompt for the required information to set up the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
- Once Terraform is detected or installed, the Terraform plan is executed
- The command prompts you with the outcome of the plan and allows you to view more details or continue with Terraform apply
- If confirmed, Terraform apply runs, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter values required for Terraform code generation.

```
lacework generate k8s eks [flags]
```

Options

```
      --apply                                     run terraform apply without executing plan or
prompting
      --aws_profile string                        specify aws profile
      --bucket_lifecycle_exp_days int             specify the s3 bucket lifecycle expiration days
      --bucket_sse_algorithm string               specify the encryption algorithm to use for S3
bucket server-side encryption
      --bucket_sse_key_arn string                 specify the kms key arn to be used for s3.
(required when bucket_sse_algorithm is aws:kms & using an existing kms key)
      --custom_filter_pattern string              specify a custom cloudwatch log filter pattern
      --enable_bucket_versioning                  enable s3 bucket versioning (default true)
      --enable_encryption_s3                      enable encryption on s3 bucket (default true)
      --enable_firehose_encryption                enable firehose encryption (default true)
      --enable_kms_key_rotation                   enable automatic kms key rotation (default true)
      --enable_mfa_delete_s3                      enable mfa delete on s3 bucket. Requires bucket
versioning.
      --enable_sns_topic_encryption               enable encryption on the sns topic (default
true)
      --existing_bucket_arn string                specify existing s3 bucket arn for the audit log
      --existing_ca_iam_role_arn string           specify existing cross account iam role arn to
use
      --existing_ca_iam_role_external_id string   specify existing cross account iam role
external_id to use
      --existing_cw_iam_role_arn string           specify existing cloudwatch iam role arn to use
      --existing_firehose_iam_role_arn string     specify existing firehose iam role arn to use
      --firehose_encryption_key_arn string        specify the kms key arn to be used with the
Firehose
  -h, --help                                      help for eks
      --integration_name string                   specify the name of the eks audit integration
      --kms_key_deletion_days int                 specify the kms waiting period before deletion,
in number of days
      --lacework_aws_account_id string            the Lacework AWS root account id
      --output string                             location to write generated content
      --prefix string                             specify the prefix that will be used at the
beginning of every generated resource
      --region_clusters stringToString            configure eks clusters per aws region. To
configure multiple regions pass the flag multiple times. Example format:  --region_clusters
<region>="cluster,list" (default [])
      --sns_topic_encryption_key_arn string       specify the kms key arn to be used with the sns
topic
      --use_existing_bucket                       use existing supplied s3 bucket (default false)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate k8s - Generate Kubernetes integration IaC

lacework generate k8s gke

Generate and/or execute Terraform code for GKE integration

Synopsis

Use this command to generate Terraform code for deploying Lacework into a GKE environment.

By default, this command interactively prompts for the required information to setup the new cloud account. In interactive mode, this command will:

- Prompt for the required information to setup the integration
- Generate new Terraform code using the inputs
- Optionally, run the generated Terraform code:
- If Terraform is already installed, the version is verified as compatible for use
- If Terraform is not installed, or the version installed is not compatible, a new version will be installed into a temporary location
  - Once Terraform is detected or installed, Terraform plan will be executed
- The command will prompt with the outcome of the plan and allow to view more details or continue with Terraform apply
  - If confirmed, Terraform apply will be run, completing the setup of the cloud account

This command can also be run in noninteractive mode. See help output for more details on the parameter value (s) required for Terraform code generation.

```
lacework generate k8s gke [flags]
```

Options

```
      --apply                                          run terraform apply without executing plan
or prompting
      --existing_service_account_name string           specify existing service account name
      --existing_service_account_private_key string    specify existing service account private key
(base64 encoded)
      --existing_sink_name string                      specify existing sink name
  -h, --help                                           help for gke
      --integration_name string                        specify a custom integration name
      --organization_id string                         specify the organization id (only set if
organization_integration is set)
      --organization_integration                       enable organization integration
      --output string                                  location to write generated content (default
is ~/lacework/gcp)
      --prefix string                                  prefix that will be used at the beginning of
every generated resource
      --project_id string                              specify the project id to be used to
provision lacework resources (required)
      --service_account_credentials string             specify service account credentials JSON
file path (leave blank to make use of google credential ENV vars)
      --wait_time string                               amount of time to wait before the next
resource is provisioned
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework generate k8s - Generate Kubernetes integration IaC

lacework policy

Manage policies

Synopsis

Manage policies in your Lacework account.

Policies add annotated metadata to queries for improving the context of alerts, reports, and information displayed in the Lacework Console.

Policies also facilitate the scheduled execution of Lacework queries.

Queries let you interactively request information from specified curated datasources. Queries have a defined structure for authoring detections.

Lacework ships a set of default LQL policies that are available in your account.

Limitations:

- The maximum number of records that each policy will return is 1000
- The maximum number of API calls is 120 per hour for on-demand LQL query executions

To view all the policies in your Lacework account.

```
lacework policy ls
```

To view more details about a single policy.

```
lacework policy show <policy_id>
```

To view the LQL query associated with the policy, use the query ID.

```
lacework query show <query_id>
```

Note: LQL syntax may change.

Options

```
  -h, --help   help for policy
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework policy create - Create a policy
- lacework policy delete - Delete a policy
- lacework policy disable - Disable policies
- lacework policy enable - Enable policies
- lacework policy list - List all policies
- lacework policy list-tags - List policy tags
- lacework policy show - Show details about a policy
- lacework policy update - Update a policy

lacework policy-exception

Manage policy exceptions

Synopsis

Manage policy exceptions in your Lacework account.

To view all the policies in your Lacework account.

```
lacework policy list
```

Options

```
  -h, --help   help for policy-exception
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework policy-exception create - Create a policy exception
- lacework policy-exception delete - Delete a policy exception
- lacework policy-exception list - List all exceptions from a single policy
- lacework policy-exception show - Show details about a policy exception

lacework policy-exception create

Create a policy exception

Synopsis

Create a new policy exception.

To create a new policy exception, run the command:

```
lacework policy-exception create [policy_id]
```

If you run the command without providing the policy_id, a list of policies is displayed in an interactive prompt.

```
lacework policy-exception create [policy_id] [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy-exception - Manage policy exceptions

lacework policy-exception delete

Delete a policy exception

Synopsis

Delete a policy exception.

To remove a policy exception, run the delete command with policy ID and exception ID arguments:

```
lacework policy-exception delete <policy_id> <exception_id>
```

```
lacework policy-exception delete <policy_id> <exception_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy-exception - Manage policy exceptions

lacework policy-exception list

List all exceptions from a single policy

Synopsis

List all of the policy exceptions from the provided policy ID.

```
lacework policy-exception list <policy_id> [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy-exception - Manage policy exceptions

lacework policy-exception show

Show details about a policy exception

Synopsis

Show the details of a policy exception.

```
lacework policy-exception show <policy_id> <exception_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy-exception - Manage policy exceptions

lacework policy create

Create a policy

Synopsis

Create a policy.

A policy is represented in either JSON or YAML format.

The following attributes are minimally required:

```
lacework policy create [flags]
```

Options

```
  -f, --file string   path to a policy to create
  -h, --help          help for create
  -u, --url string    url to a policy to create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
```

```
      --api_token string   access token (replaces the use of api_key and api_secret)
      --debug              turn on debug logging
      --json               switch commands output from human-readable to json format
      --nocache            turn off caching
      --nocolor            turn off colors
      --noninteractive     turn off interactive mode (disable spinners, prompts, etc.)
      --organization       access organization level data sets (org admins only)
  -p, --profile string     switch between profiles configured at ~/.lacework.toml
      --subaccount string  sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy delete

Delete a policy

Synopsis

Delete a policy by providing the policy ID.

Use the command 'lacework policy list' to list the registered policies in your Lacework account.

```
lacework policy delete <policy_id> [flags]
```

Options

```
      --cascade   delete policy and its associated query
  -h, --help      help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy disable

Disable policies

Synopsis

Disable policies by ID or all policies matching a tag.

To disable a single policy by its ID:

```
lacework policy disable lacework-policy-id
```

To disable many policies by ID provide a list of policy ids:

```
lacework policy disable lacework-policy-id-one lacework-policy-id-two
```

To disable all policies for AWS CIS 1.4.0:

```
lacework policy disable --tag framework:cis-aws-1-4-0
```

To disable all policies for GCP CIS 1.3.0:

```
lacework policy disable --tag framework:cis-gcp-1-3-0
```

```
lacework policy disable [policy_id...] [flags]
```

Options

```
  -h, --help         help for disable
      --tag string   disable all policies with the specified tag
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy enable

Enable policies

Synopsis

Enable policies by ID or all policies matching a tag.

To enter the policy enable prompt:

```
lacework policy enable
```

To enable a single policy by its ID:

```
lacework policy enable lacework-policy-id
```

To enable many policies by ID provide a list of policy ids:

```
lacework policy enable lacework-policy-id-one lacework-policy-id-two
```

To enable all policies for AWS CIS 1.4.0:

```
lacework policy enable --tag framework:cis-aws-1-4-0
```

To enable all policies for GCP CIS 1.3.0:

```
lacework policy enable --tag framework:cis-gcp-1-3-0
```

```
lacework policy enable [policy_id...] [flags]
```

Options

```
  -h, --help         help for enable
      --tag string   enable all policies with the specified tag
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy list

List all policies

Synopsis

List all registered policies in your Lacework account.

```
lacework policy list [flags]
```

Options

```
      --alert_enabled     only show alert_enabled policies
      --enabled           only show enabled policies
  -h, --help              help for list
      --severity string   filter policies by severity threshold (critical, high, medium, low, 
info)
      --tag string        only show policies with the specified tag
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy list-tags

List policy tags

Synopsis

List all tags associated with policies in your Lacework account.

```
lacework policy list-tags [flags]
```

Options

```
  -h, --help   help for list-tags
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy show

Show details about a policy

Synopsis

Show details about the provided policy ID.

```
lacework policy show <policy_id> [flags]
```

Options

```
  -h, --help   help for show
      --yaml   output query in YAML format
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework policy update

Update a policy

Synopsis

Update a policy.

A policy identifier is required to update a policy.

A policy identifier can be specified via:

1. A policy update command argument
   lacework policy update my-policy-1
2. The policy update payload
   { "policy_id": "my-policy-1", "severity": "critical" }

A policy identifier specified via command argument always takes precedence over a policy identifer specified via payload.

The severity of many policies can be updated at once by passing a list of policy identifiers:

```
lacework policy update my-policy-1 my-policy-2 --severity critical
```

```
lacework policy update [policy_id...] [flags]
```

Options

```
  -f, --file string       path to a policy to update
  -h, --help              help for update
      --severity string   update the policy severity
  -u, --url string        url to a policy to update
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework policy - Manage policies

lacework query

Run and manage queries

Synopsis

Run and manage Lacework Query Language (LQL) queries.

LQL is a SQL-like query language for specifying the selection, filtering, and manipulation of data. Queries let you interactively request information from specified curated datasources.

Lacework ships a set of default LQL queries that are available in your account.

For more information about LQL, visit:

```
https://docs.lacework.com/lql-overview
```

To view all LQL queries in your Lacework account.

```
lacework query ls
```

To show a query.

```
lacework query show <query_id>
```

To execute a query.

```
lacework query run <query_id>
```

Note: LQL syntax may change.

Options

```
  -h, --help   help for query
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework query create - Create a query
- lacework query delete - Delete a query
- lacework query list - List queries
- lacework query list-sources - List Lacework query datasources
- lacework query preview-source - Preview Lacework query datasource
- lacework query run - Run a query
- lacework query show - Show a query
- lacework query show-source - Show Lacework query datasource

- lacework query update - Update a query
- lacework query validate - Validate a query

lacework query create

Create a query

Synopsis

There are multiple ways you can create a query:

- Typing the query into your default editor (via $EDITOR)
- Piping a query to the Lacework CLI command (via $STDIN)
- From a local file on disk using the flag '--file'
- From a URL using the flag '--url'

There are also multiple formats you can use to define a query:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and create a new query.

```
lacework lql create
```

The following example checks for unrestricted ingress to TCP port 445:

A query is represented using JSON or YAML markup and must specify both 'queryId' and 'queryText' keys. The above query uses YAML, specifies an identifier of 'LW_Custom_UnrestrictedIngressToTCP445', and identifies AWS EC2 security groups with unrestricted access to TCP port 445. The queryText is expressed in Lacework Query Language (LQL) syntax which is delimited by '{ }' and contains three sections:

- Source data is specified in the 'source' clause. The source of data is the 'LW_CFG_AWS_EC2_SECURITY_GROUPS' datasource. LQL queries generally refer to other datasources, and customizable policies always target a suitable datasource.
- Records of interest are specified by the 'filter' clause. In the example, the records available in 'LW_CFG_AWS_EC2_SECURITY_GROUPS' are filtered for those whose IP protocol is 'tcp', whose from and to port is '445', and CidrIP is '0.0.0.0/0'. The syntax for this filtering expression strongly resembles SQL.
- The fields this query exposes are listed in the 'return' clause. Because there may be unwanted duplicates among result records when Lacework composes them from just these four columns, the distinct modifier is added. This behaves like a SQL 'SELECT DISTINCT'. Each returned column in this case is just a field that is present in 'LW_CFG_AWS_EC2_SECURITY_GROUPS', but you can compose results by manipulating strings, dates, JSON and numbers as well.

The resulting dataset is shaped like a table. The table's columns are named with the names of the columns selected. If desired, you could alias them to other names as well.

For more information about LQL, visit:

```
https://docs.lacework.com/lql-overview
```

```
lacework query create [flags]
```

Options

```
  -f, --file string   path to a query to create
  -h, --help          help for create
  -u, --url string    url to a query to create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query delete

Delete a query

Synopsis

Delete a single LQL query by providing the query ID.

Use the command 'lacework query list' to list the available queries in your Lacework account.

```
lacework query delete <query_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query list

List queries

Synopsis

List all LQL queries in your Lacework account.

```
lacework query list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query list-sources

List Lacework query datasources

Synopsis

List Lacework query datasources.

```
lacework query list-sources [flags]
```

Options

```
  -h, --help   help for list-sources
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query preview-source

Preview Lacework query datasource

Synopsis

Preview Lacework query datasource.

```
lacework query preview-source <datasource_id> [flags]
```

Options

```
  -h, --help   help for preview-source
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query run

Run a query

Synopsis

Run an LQL query via editor:

```
lacework query run --range today
```

Run a query via ID (uses active profile):

```
lacework query run MyQuery --start "-1w@w" --end "@w"
```

Start and end times are required to run a query:

1. Specify start and end times in one of the following formats:
   A. A relative time specifier B. RFC3339 date and time C. Epoch time in milliseconds
2. Specify start and end times in one of the following ways:
   A. As StartTimeRange and EndTimeRange in the ParamInfo block within the query B. As start_time_range and end_time_range if specifying JSON C. As --start and --end CLI flags
3. Start and End time precedence:
   A. CLI flags take precedence over JSON specifications

```
lacework query run [query_id] [flags]
```

Options

```
      --empty                  start $EDITOR with empty file
      --end string             end time for query (default "now")
      --fail_on_count string   fail if the results from a query match the provided expression
(e.g. '>0')
  -f, --file string            path to a query to run
  -h, --help                   help for run
      --limit int              result limit for query (default 0)
      --range string           natural time range for query
      --start string           start time for query (default "-24h")
  -u, --url string             url to a query to run
      --validate_only          validate query only (do not run)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query show

Show a query

Synopsis

Show a query in your Lacework account.

```
lacework query show <query_id> [flags]
```

Options

```
  -h, --help   help for show
      --yaml   output query in YAML format
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query show-source

Show Lacework query datasource

Synopsis

Show Lacework query datasource.

```
lacework query show-source <datasource_id> [flags]
```

Options

```
  -h, --help   help for show-source
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query update

Update a query

Synopsis

There are multiple ways you can update a query:

- Typing the query into your default editor (via $EDITOR)
- Passing a query ID to load it into your default editor
- From a local file on disk using the flag '--file'
- From a URL using the flag '--url'

There are also multiple formats you can use to define a query:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and update a query.

```
lacework query update
```

```
lacework query update [query_id] [flags]
```

Options

```
  -f, --file string   path to a query to update
  -h, --help          help for update
  -u, --url string    url to a query to update
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework query validate

Validate a query

Synopsis

Use this command to validate a single LQL query before creating it.

There are multiple ways you can validate a query:

- Typing the query into your default editor (via $EDITOR)
- From a local file on disk using the flag '--file'
- From a URL using the flag '--url'

There are also multiple formats you can use to define a query:

- Javascript Object Notation (JSON)
- YAML Ain't Markup Language (YAML)

To launch your default editor and validate a query.

```
lacework query validate
```

```
lacework query validate [flags]
```

Options

```
  -f, --file string   path to a query to validate
  -h, --help          help for validate
  -u, --url string    url to a query to validate
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework query - Run and manage queries

lacework report-rule

Manage report rules

Synopsis

Manage report rules to route reports to one or more email alert channels.

A report rule has four parts:

```
1. Email alert channel(s) that should receive the report
2. One or more severities to include
3. Resource group(s) containing the subset of your environment to consider
4. Notification types containing which report information to send
```

Options

```
  -h, --help   help for report-rule
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework report-rule create - Create a new report rule
- lacework report-rule delete - Delete a report rule
- lacework report-rule list - List all report rules
- lacework report-rule show - Show a report rule by ID

lacework report-rule create

Create a new report rule

```
lacework report-rule create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework report-rule - Manage report rules

lacework report-rule delete

Delete a report rule

Synopsis

Delete a single report rule by it's ID.

```
lacework report-rule delete <report_rule_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework report-rule - Manage report rules

lacework report-rule list

List all report rules

Synopsis

List all report rules configured in your Lacework account.

```
lacework report-rule list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework report-rule - Manage report rules

lacework report-rule show

Show a report rule by ID

Synopsis

Show a single report rule by it's ID.

```
lacework report-rule show <report_rule_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework report-rule - Manage report rules

lacework resource-group

Manage resource groups

Synopsis

Manage Lacework-identifiable assets via the use of resource groups.

Options

```
-h, --help   help for resource-group
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
```

```
      --json                 switch commands output from human-readable to json format
      --nocache              turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework resource-group create - Create a new resource group
- lacework resource-group delete - Delete a resource group
- lacework resource-group list - List all resource groups
- lacework resource-group show - Get resource group by ID

lacework resource-group create

Create a new resource group

Synopsis

Creates a new single resource group.

```
lacework resource-group create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework resource-group - Manage resource groups

lacework resource-group delete

Delete a resource group

Synopsis

Delete a single resource group by it's resource group ID.

```
lacework resource-group delete <resource_group_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework resource-group - Manage resource groups

lacework resource-group list

List all resource groups

Synopsis

List all resource groups configured in your Lacework account.

```
lacework resource-group list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework resource-group - Manage resource groups

lacework resource-group show

Get resource group by ID

Synopsis

Get a single resource group by it's resource group ID.

```
lacework resource-group show <resource_group_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework resource-group - Manage resource groups

lacework team-member

Manage team members

Synopsis

Manage Team Members to grant or restrict access to multiple Lacework Accounts. Team members can also be granted organization-level roles.

Options

```
  -h, --help   help for team-member
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework team-member create - Create a new team member
- lacework team-member delete - Delete a team member
- lacework team-member list - List all team members
- lacework team-member show - Show a team member by id

lacework team-member create

Create a new team member

```
lacework team-member create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework team-member - Manage team members

lacework team-member delete

Delete a team member

Synopsis

Delete a single team member by it's ID.

```
lacework team-member delete <team_member_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework team-member - Manage team members

lacework team-member list

List all team members

Synopsis

List all team members configured in your Lacework account.

```
lacework team-member list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework team-member - Manage team members

lacework team-member show

Show a team member by id

Synopsis

Show a single team member by it's id.

```
lacework team-member show <team_member_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
```

```
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework team-member - Manage team members

lacework version

Print the Lacework CLI version

Synopsis

Prints out the installed version of the Lacework CLI and checks for newer versions available for update.

Set the environment variable 'LW_UPDATES_DISABLE=1' to avoid checking for updates.

```
lacework version [flags]
```

Options

```
-h, --help   help for version
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.

lacework vulnerability

Container and host vulnerability assessments

Synopsis

Container and host vulnerability assessments.

Options

```
  -h, --help   help for vulnerability
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework vulnerability container - Vulnerability assessment for containers
- lacework vulnerability host - Vulnerability assessment for hosts

lacework vulnerability-exception

Manage vulnerability exceptions

Synopsis

Manage vulnerability exceptions to control and customize your alert profile for hosts and containers.

Options

```
  -h, --help   help for vulnerability-exception
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework - A tool to manage the Lacework cloud security platform.
- lacework vulnerability-exception create - Create a new vulnerability exception
- lacework vulnerability-exception delete - Delete a vulnerability exception
- lacework vulnerability-exception list - List all vulnerability exceptions
- lacework vulnerability-exception show - Get vulnerability exception by ID

lacework vulnerability-exception create

Create a new vulnerability exception

Synopsis

Creates a new single vulnerability exception.

```
lacework vulnerability-exception create [flags]
```

Options

```
  -h, --help   help for create
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability-exception - Manage vulnerability exceptions

lacework vulnerability-exception delete

Delete a vulnerability exception

Synopsis

Delete a single vulnerability exception by it's vulnerability exception ID.

```
lacework vulnerability-exception delete <exception_id> [flags]
```

Options

```
  -h, --help   help for delete
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability-exception - Manage vulnerability exceptions

lacework vulnerability-exception list

List all vulnerability exceptions

Synopsis

List all vulnerability exceptions configured in your Lacework account.

```
lacework vulnerability-exception list [flags]
```

Options

```
  -h, --help   help for list
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability-exception - Manage vulnerability exceptions

lacework vulnerability-exception show

Get vulnerability exception by ID

Synopsis

Get a single vulnerability exception by it's vulnerability exception ID.

```
lacework vulnerability-exception show <exception_id> [flags]
```

Options

```
  -h, --help   help for show
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability-exception - Manage vulnerability exceptions

lacework vulnerability container

Vulnerability assessment for containers

Synopsis

Request on-demand container vulnerability scans and show previous assessments from published images.

PREREQUISITE: Your Lacework account should already be configured with a Container Registry Integration of the container images you are trying to scan or show.

To create a new integration use the following command:

```
lacework container-registry create
```

If you prefer to configure the integration via the WebUI, log in to your account at:

```
https://<ACCOUNT>.lacework.net
```

Then navigate to Settings > Integrations > Container Registry.

Options

```
  -h, --help   help for container
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability - Container and host vulnerability assessments
- lacework vulnerability container list-assessments - List container vulnerability assessments (default last 24 hours)
- lacework vulnerability container list-registries - List all container registries configured
- lacework vulnerability container scan - Request an on-demand container vulnerability assessment
- lacework vulnerability container show-assessment - Show results of a container vulnerability assessment

lacework vulnerability container list-assessments

List container vulnerability assessments (default last 24 hours)

Synopsis

List all container vulnerability assessments for the last 24 hours by default.

To customize the time range use use '--start', '--end', or '--range'.

The start and end times can be specified in one of the following formats:

```
A. A relative time specifier
B. RFC3339 date and time
C. Epoch time in milliseconds
```

Or use a natural time range like.

```
lacework vuln container list --range yesterday
```

The natural time range of 'yesterday' would represent a relative start time of '-1d@d' and a relative end time of '@d'.

You can also pass '--fixable' to filter on containers with vulnerabilities that have fixes available, or '--active' to filter on container images actively running in your environment.

```
lacework vulnerability container list-assessments [flags]
```

Options

```
      --active               only show vulnerabilities of packages actively running in your
environment
      --csv                  output vulnerability assessment in CSV format
      --end string           end of the time range (default "now")
      --fixable              only show fixable vulnerabilities
  -h, --help                 help for list-assessments
      --range string         natural time range for query
      --registry strings     filter assessments for specific registries
  -r, --repository strings   filter assessments for specific repositories
      --start string         start of the time range (default "-24h")
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability container - Vulnerability assessment for containers

lacework vulnerability container list-registries

List all container registries configured

Synopsis

List all container registries configured in your account.

```
lacework vulnerability container list-registries [flags]
```

Options

```
  -h, --help   help for list-registries
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability container - Vulnerability assessment for containers

lacework vulnerability container scan

Request an on-demand container vulnerability assessment

Synopsis

Request on-demand container vulnerability assessments and view the generated results.

To list all container registries configured in your account:

```
lacework vulnerability container list-registries
```

NOTE: Scans can take up to 15 minutes to return results.

Arguments:

```
<registry>    container registry where the container image has been published
<repository>  repository name that contains the container image
<tag|digest>  either a tag or an image digest to scan (digest format: sha256:1ee...1d3b)
```

```
lacework vulnerability container scan <registry> <repository> <tag|digest> [flags]
```

Options

```
      --details                   increase details of a vulnerability assessment
      --fail_on_fixable           fail if the assessed container has fixable vulnerabilities
      --fail_on_severity string   specify a severity threshold to fail if vulnerabilities are
found (critical, high, medium, low, info)
      --fixable                   only show fixable vulnerabilities
  -h, --help                      help for scan
      --html                      generate a vulnerability assessment in HTML format
      --packages                  show a list of packages with CVE count
      --poll                      poll until the vulnerability scan completes
      --severity string           filter vulnerability assessment by severity threshold (critical,
high, medium, low, info)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache              turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability container - Vulnerability assessment for containers

lacework vulnerability container show-assessment

Show results of a container vulnerability assessment

Synopsis

Show the vulnerability assessment results of the specified container.

Arguments:

```
<sha256:hash> a sha256 hash of a container image (format: sha256:1ee...1d3b)
```

Note that the provided SHA is treated first as the image digest, but if no results are found, this commands tries to use the SHA as the image id.

To request an on-demand vulnerability scan:

```
lacework vulnerability container scan <registry> <repository> <tag|digest>
```

To see details for a single cve result in an assessment:

```
lacework vulnerability show-assessment <sha256:hash> [cve_id]
```

```
lacework vulnerability container show-assessment <sha256:hash> [cve_id] [flags]
```

Options

```
      --csv                        output vulnerability assessment in CSV format
      --details                    increase details of a vulnerability assessment
      --fail_on_fixable          fail if the assessed container has fixable vulnerabilities
      --fail_on_severity string   specify a severity threshold to fail if vulnerabilities are
found (critical, high, medium, low, info)
      --fixable                  only show fixable vulnerabilities
  -h, --help                     help for show-assessment
      --html                     generate a vulnerability assessment in HTML format
      --packages                 show a list of packages with CVE count
      --severity string          filter vulnerability assessment by severity threshold (critical,
high, medium, low, info)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability container - Vulnerability assessment for containers

lacework vulnerability host

Vulnerability assessment for hosts

Synopsis

Request on-demand host vulnerability scans and show previous assessments from hosts with the Lacework datacollector agent installed.

Options

```
-h, --help   help for host
```

Options inherited from parent commands

```
-a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
-k, --api_key string      access key id
-s, --api_secret string   secret access key
    --api_token string    access token (replaces the use of api_key and api_secret)
    --debug               turn on debug logging
    --json                switch commands output from human-readable to json format
    --nocache             turn off caching
    --nocolor             turn off colors
    --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
    --organization        access organization level data sets (org admins only)
-p, --profile string      switch between profiles configured at ~/.lacework.toml
    --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability - Container and host vulnerability assessments
- lacework vulnerability host generate-pkg-manifest - Generates a package-manifest from the local host
- lacework vulnerability host list-cves - List the CVEs found in the hosts in your environment
- lacework vulnerability host list-hosts - List the hosts that contain a specified CVE ID in your environment
- lacework vulnerability host scan-pkg-manifest - Request an on-demand host vulnerability assessment from a package-manifest
- lacework vulnerability host show-assessment - Show results of a host vulnerability assessment

lacework vulnerability host generate-pkg-manifest

Generates a package-manifest from the local host

Synopsis

Generates a package-manifest formatted for usage with the Lacework scan package-manifest API.

Additionally, you can automatically generate a package-manifest from the local host and send it directly to the Lacework API with the command:

```
lacework vulnerability host scan-pkg-manifest --local
```

```
lacework vulnerability host generate-pkg-manifest [flags]
```

Options

```
  -h, --help   help for generate-pkg-manifest
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability host - Vulnerability assessment for hosts

lacework vulnerability host list-cves

List the CVEs found in the hosts in your environment

Synopsis

List the CVEs found in the hosts in your environment.

Filter results to only show vulnerabilities actively running in your environment with fixes:

```
lacework vulnerability host list-cves --active --fixable
```

```
lacework vulnerability host list-cves [flags]
```

Options

```
      --active            only show vulnerabilities of packages actively running in your
environment
      --csv               output vulnerability assessment in CSV format
      --end string        end of the time range (default "now")
      --fixable           only show fixable vulnerabilities
  -h, --help              help for list-cves
      --packages          show a list of packages with CVE count
      --range string      natural time range for query
      --severity string   filter vulnerability assessment by severity threshold (critical, high,
medium, low, info)
      --start string      start of the time range (default "-24h")
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability host - Vulnerability assessment for hosts

lacework vulnerability host list-hosts

List the hosts that contain a specified CVE ID in your environment

Synopsis

List the hosts that contain a specified CVE ID in your environment.

To list the CVEs found in the hosts of your environment run:

```
lacework vulnerability host list-cves
```

```
lacework vulnerability host list-hosts <cve_id> [flags]
```

Options

```
      --csv            output vulnerability assessment in CSV format
      --end string     end of the time range (default "now")
  -h, --help           help for list-hosts
      --range string   natural time range for query
      --start string   start of the time range (default "-24h")
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability host - Vulnerability assessment for hosts

lacework vulnerability host scan-pkg-manifest

Request an on-demand host vulnerability assessment from a package-manifest

Synopsis

Request an on-demand host vulnerability assessment of your software packages to determine if the packages contain any common vulnerabilities and exposures.

Simple usage:

```
lacework vulnerability host scan-pkg-manifest '{
    "osPkgInfoList": [
        {
            "os":"Ubuntu",
            "osVer":"18.04",
            "pkg": "openssl",
            "pkgVer": "1.1.1-1ubuntu2.1~18.04.5"
        }
    ]
}'
```

To generate a package-manifest from the local host and scan it automatically:

```
lacework vulnerability host scan-pkg-manifest --local
```

NOTE: Only packages managed by a package manager for supported operating systems are reported. **NOTE:** This operation is limited to 10k packages per command execution.

```
lacework vulnerability host scan-pkg-manifest <manifest> [flags]
```

Options

```
      --fail_on_fixable           fail if the assessed container has fixable vulnerabilities
      --fail_on_severity string   specify a severity threshold to fail if vulnerabilities are
found (critical, high, medium, low, info)
  -f, --file string               path to a package manifest to scan
      --fixable                   only show fixable vulnerabilities
  -h, --help                      help for scan-pkg-manifest
  -l, --local                     automatically generate the package manifest from the local host
      --packages                  show a list of packages with CVE count
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor              turn off colors
      --noninteractive       turn off interactive mode (disable spinners, prompts, etc.)
      --organization         access organization level data sets (org admins only)
  -p, --profile string       switch between profiles configured at ~/.lacework.toml
      --subaccount string    sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability host - Vulnerability assessment for hosts

lacework vulnerability host show-assessment

Show results of a host vulnerability assessment

Synopsis

Show results of a host vulnerability assessment.

To find the machine id from hosts in your environment, use the command:

```
lacework vulnerability host list-cves
```

Grab a CVE id and feed it to the command:

```
lacework vulnerability host list-hosts my_cve_id
```

```
lacework vulnerability host show-assessment <machine_id> [flags]
```

Options

```
      --active                     only show vulnerabilities of packages actively running in your
environment
      --collector_type string      filter assessments by collector type (Agent or Agentless)
(default "Agentless")
      --csv                        output vulnerability assessment in CSV format
      --details                    increase details of a vulnerability assessment
      --fail_on_fixable            fail if the assessed container has fixable vulnerabilities
      --fail_on_severity string    specify a severity threshold to fail if vulnerabilities are
found (critical, high, medium, low, info)
      --fixable                    only show fixable vulnerabilities
  -h, --help                       help for show-assessment
      --packages                   show a list of packages with CVE count
      --severity string           filter vulnerability assessment by severity threshold (critical,
high, medium, low, info)
```

Options inherited from parent commands

```
  -a, --account string      account subdomain of URL (i.e. <ACCOUNT>.lacework.net)
  -k, --api_key string      access key id
  -s, --api_secret string   secret access key
      --api_token string    access token (replaces the use of api_key and api_secret)
      --debug               turn on debug logging
      --json                switch commands output from human-readable to json format
      --nocache             turn off caching
      --nocolor             turn off colors
      --noninteractive      turn off interactive mode (disable spinners, prompts, etc.)
      --organization        access organization level data sets (org admins only)
  -p, --profile string      switch between profiles configured at ~/.lacework.toml
      --subaccount string   sub-account name inside your organization (org admins only)
```

See also

- lacework vulnerability host - Vulnerability assessment for hosts www.fortinet.com