

Integrating AWS

Updated on 04 Aug 2024 · 1 Minute to read · Contributors 

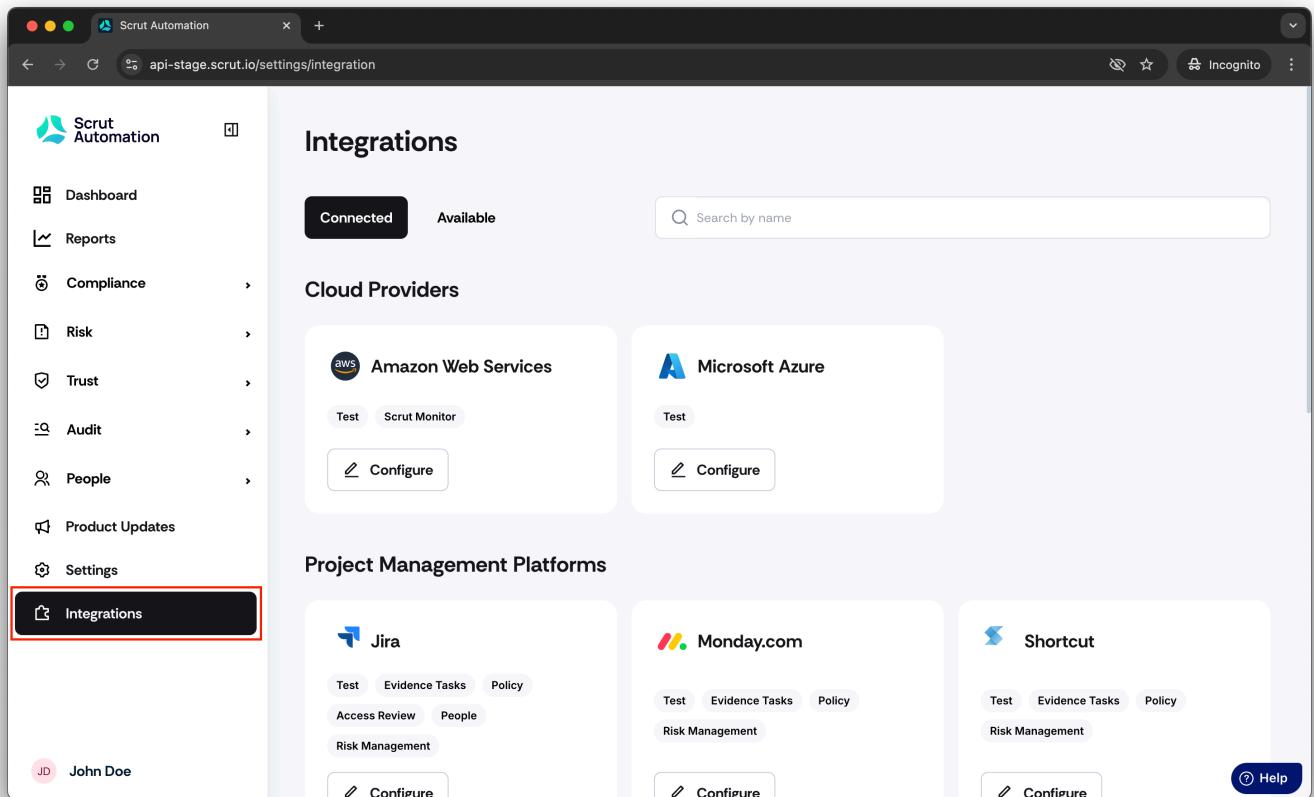
Amazon Web Services (AWS) is a widely used cloud provider. Scrut allows users to integrate one or more AWS accounts for automated daily scans of misconfigurations, with results displayed in the [Cloud -> Test module](#) of Scrut.

Contents:

1. [Integrating first AWS account](#)
2. [Integrating multiple AWS accounts](#)
3. [AWS CodeCommit Integration: Auto-Collect Pull Request Evidence](#)

Integrating first AWS account

1. From the left navigation panel, click **Integrations**.



The screenshot shows the Scrut Automation web application. The left sidebar has a 'Dashboard' icon, followed by a collapsed 'Reports' section, then 'Compliance', 'Risk', 'Trust', 'Audit', 'People', 'Product Updates', 'Settings', and finally 'Integrations' which is highlighted with a red box. The main content area is titled 'Integrations'. It has two tabs: 'Connected' (selected) and 'Available'. A search bar says 'Search by name'. Below are sections for 'Cloud Providers' and 'Project Management Platforms'. In 'Cloud Providers', there are cards for 'Amazon Web Services' (with 'Test' and 'Scrut Monitor' buttons) and 'Microsoft Azure' (with 'Test' button). In 'Project Management Platforms', there are cards for 'Jira' (with 'Test', 'Evidence Tasks', 'Policy', 'Access Review', 'People', and 'Risk Management' buttons), 'Monday.com' (with 'Test', 'Evidence Tasks', 'Policy', and 'Risk Management' buttons), and 'Shortcut' (with 'Test', 'Evidence Tasks', 'Policy', and 'Risk Management' buttons). Each card has a 'Configure' button at the bottom right.

2. Click Available.

The screenshot shows the 'Integrations' page of the Scrut Automation web application. On the left, a sidebar lists various sections: Dashboard, Reports, Compliance, Risk, Trust, Audit, People, Product Updates, Settings, and Integrations. The 'Integrations' section is currently selected and highlighted with a dark background. At the top center, there are two buttons: 'Connected' and 'Available'. The 'Available' button is highlighted with a red border. To the right of the buttons is a search bar with the placeholder 'Search by name'. In the bottom left corner of the main area, there is a small user profile icon with the initials 'JD' and the name 'John Doe'. In the bottom right corner, there is a blue button with a help icon and the word 'Help'.

3. Go to Cloud Provider section and find AWS.

4. Click Integrate.

The screenshot shows the 'Integrations' page of the Scrut Automation web application. On the left, a sidebar menu includes 'Dashboard', 'Reports', 'Compliance', 'Risk', 'Trust', 'Audit', 'People', 'Product Updates', 'Settings', and 'Integrations'. The 'Integrations' item is highlighted with a dark background. The main content area is titled 'Integrations' and shows a 'Connected' tab and an 'Available' tab. A search bar at the top right says 'Search by name'. Below this, there are two sections: 'Cloud Providers' and 'Identity Providers'. The 'Cloud Providers' section contains cards for Amazon Web Services (AWS), Microsoft Azure, Google Cloud, Digital Ocean, and Heroku. Each card has a 'Test' button, a 'Scrut Monitor' button, and an 'Integrate' button. The AWS card's 'Integrate' button is highlighted with a red border. The 'Identity Providers' section is partially visible below. At the bottom right of the main area is a 'Help' button.

5. In *Account Nickname* field, enter a nickname for future reference.

The screenshot shows the 'Amazon Web Services' integration configuration page. The sidebar on the left is identical to the previous screenshot. The main area starts with a 'Not Connected' status message. Below it, there's a 'Steps To Integrate' section. The 'Credentials' section contains fields for 'Account Nickname *' and 'Role ARN *'. Both fields have placeholder text 'Enter the Account Nickname' and 'Enter the Role ARN' respectively, with character count indicators '0/100' next to them. A 'Submit' button is located below these fields. At the bottom of the page is an 'Audit Logs' section with a search bar and filter options: 'Sort By', 'Action', 'User', 'Date Filter', and 'Help'.

To get the **Role ARN**, open a new tab and login to your AWS account.

6. Click the below **CloudFormation template** link.

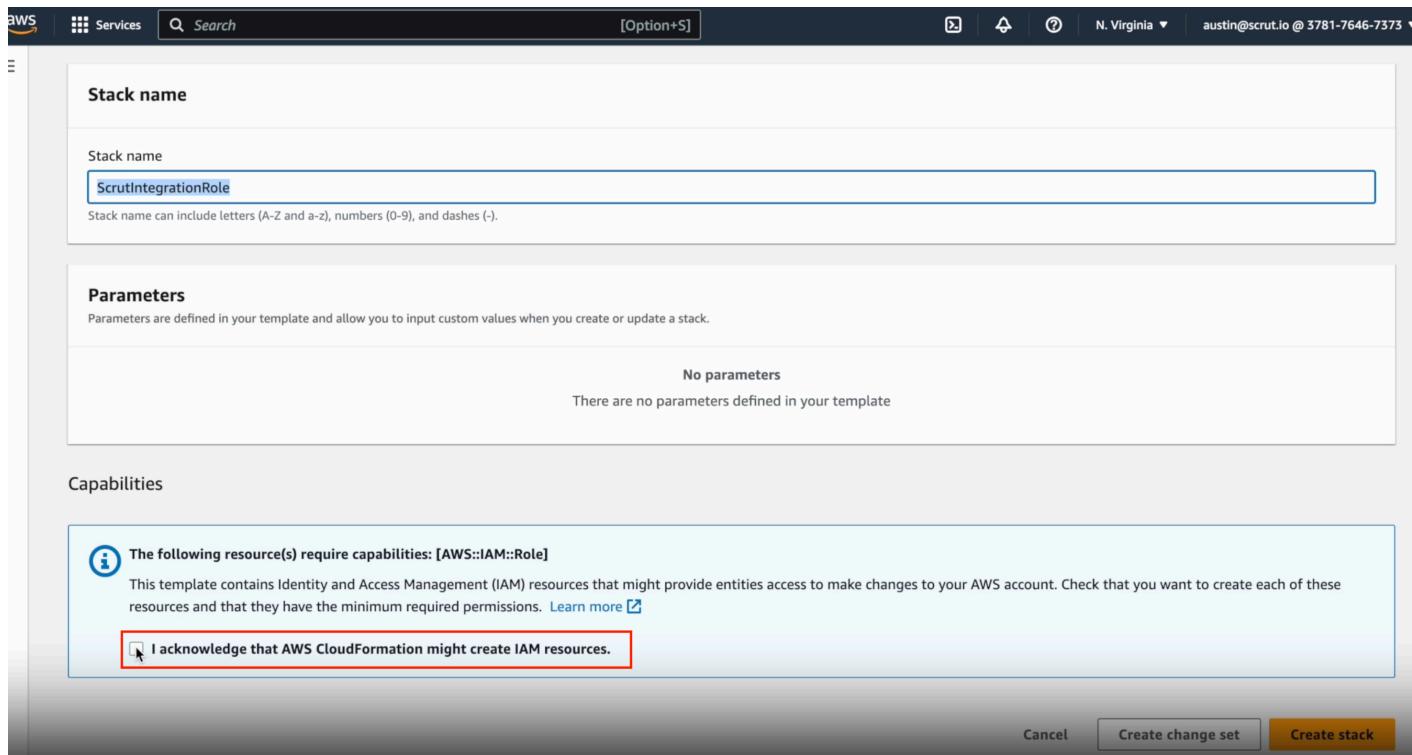
<https://us-east-1.console.aws.amazon.com/cloudformation/home#/stacks/create/review?templateURL=https://scrut-cloudformation-templates.s3.ap-south-1.amazonaws.com/iam-role-based-access.yaml&stackName=ScrutIntegrationRole>

This will redirect to the AWS console *Quick create stack* page.

The screenshot shows the AWS CloudFormation 'Create stack' interface. In the 'Template' section, the URL `https://scrut-cloudformation-templates.s3.ap-south-1.amazonaws.com/iam-role-based-access.yaml` is entered. The 'Stack name' field contains the value `ScrutIntegrationRole`. The 'Parameters' section indicates there are no parameters defined in the template.

8. In *Stack Name* field, enter value "ScrutIntegrationRole".

9. Select *IAM permissions* check box to acknowledge that AWS Cloudformation template might create IAM resources.



Stack name

Stack name

ScrutIntegrationRole

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

Capabilities

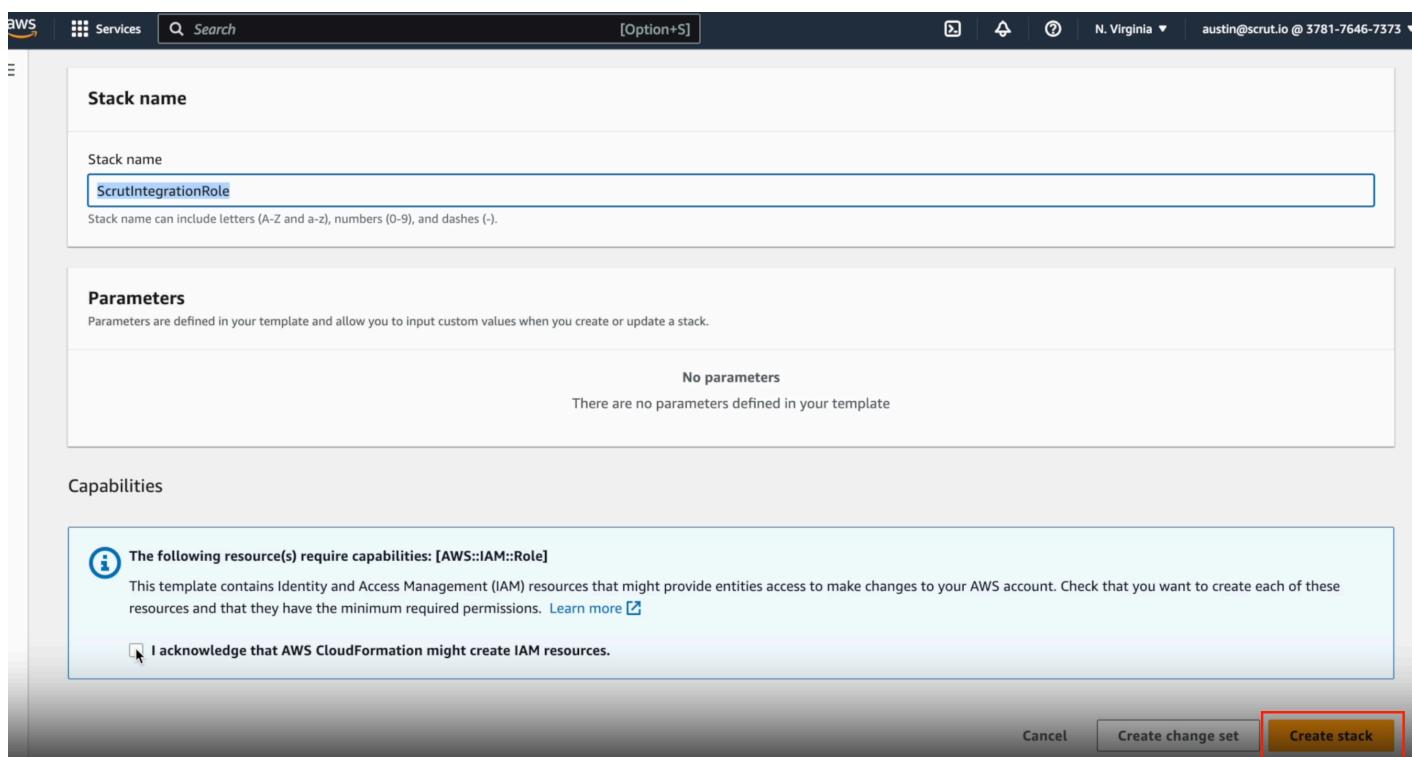
The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

I acknowledge that AWS CloudFormation might create IAM resources.

Cancel Create change set Create stack

10. Click **Create Stack**.



Stack name

Stack name

ScrutIntegrationRole

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

Capabilities

The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

I acknowledge that AWS CloudFormation might create IAM resources.

Cancel Create change set Create stack

11. Once the stack is created, open the stack go to outputs and copy the ARN value.

12. Paste it in the Scrut integrations page where it ask to enter the **Role ARN** details.

The screenshot shows the 'Integrations' section of the Scrut Automation interface. On the left sidebar, 'Integrations' is selected. In the main area, the 'Amazon Web Services' integration is being configured. The 'Credentials' step is active. The 'Role ARN' field contains the value 'arn:aws:iam::087141211690:role/ScrutIntegrationRole-Sandbox-ScrutRole-WRI2VHXN5odM'. The 'Submit' button is located at the bottom of the form.

14. Click **Submit**.

The account has been successfully integrated.

Note: It takes atmost 24 hours to populate the first scan results.

Tip: Incase you don't see the scan results, please check the Audit Logs section.

Integrating multiple AWS accounts

1. Click **Configure**.

The screenshot shows the 'Integrations' page in the Scrut Automation web application. On the left, a sidebar menu includes 'Dashboard', 'Reports', 'Compliance', 'Risk', 'Trust', 'Audit', 'People', 'Product Updates', 'Settings', and 'Integrations'. The 'Integrations' item is highlighted with a black background. A user profile for 'John Doe' is at the bottom. The main content area has two sections: 'Cloud Providers' and 'Project Management Platforms'. Under 'Cloud Providers', there are cards for 'Amazon Web Services' (with 'Configure' button) and 'Microsoft Azure' (with 'Test' button). Under 'Project Management Platforms', there are cards for 'Jira', 'Monday.com', and 'Shortcut' (each with a 'Configure' button). A search bar at the top right says 'Search by name'.

2. Click Add.

The screenshot shows the 'Amazon Web Services' integration configuration page. The sidebar menu is identical to the previous screen. The main area shows the 'Amazon Web Services' card with a green 'Connected' status indicator. Below it is a table with one row of data:

Integration ID	Status	Account Nickname	Entity	Actions
3a9fb16-04a6-4dc3-a26b-9456e8b34dd3	Active	Scrut Test		

A red box highlights the 'Add' button in the top right corner of the table header.

3. Rest of the steps are same as [Integrating first AWS account](#).

AWS CodeCommit Integration: Auto-Collect Pull Request Evidence

The AWS CodeCommit integration enhancement allows auto-collection of pull request evidence, improving Scrut's capabilities to monitor CodeCommit repositories.

Key Features:

- Repository Insights: Detailed visibility into repository creation dates, last modified dates, and default branches.
- Branch Management: Track branches, commit IDs, and branch names.
- Commit Tracking: Monitor commit details, including committer and author information, commit messages, and parent commits.
- Automated Evidence Collection:
 - List of CodeCommit repositories.
 - Detailed commit logs.
 - Status updates and details of pull requests.
 - Identification of pull request approvers.

Configuring AWS CodeCommit Integration

1. Navigate to the integrations section within Scrut.
2. Find AWS integration and Click on "Configure".

The screenshot shows the 'Integrations' page in the Scrut Automation web application. On the left, a sidebar menu includes 'Dashboard', 'Reports', 'Compliance', 'Risk', 'Trust', 'Audit', 'People', 'Product Updates', 'Settings', and 'Integrations'. The 'Integrations' item is highlighted with a dark background. A user profile for 'John Doe' is at the bottom. The main content area is titled 'Integrations' and shows two sections: 'Cloud Providers' and 'Project Management Platforms'. In 'Cloud Providers', there are cards for 'Amazon Web Services' (with 'Configure' button) and 'Microsoft Azure' (with 'Test' button). In 'Project Management Platforms', there are cards for 'Jira', 'Monday.com', and 'Shortcut' (each with a 'Configure' button). A search bar at the top right says 'Search by name'.

3. Select the specific AWS integration you want to configure for CodeCommit.

The screenshot shows the 'Amazon Web Services' integration configuration page. The sidebar menu is identical to the previous screen. The main area is titled '[← Amazon Web Services](#)'. It features a card for 'Amazon Web Services' with 'Connected' status and 'Steps To Integrate' link. Below is a table with columns: 'Integration' (selected), 'Tags', and 'Audit Logs'. The table has a search bar and filters for 'Entities', 'Status', and 'Actions'. One row is shown: 'Integration ID' is '3a9fbdb16-04a6-4dc3-a26b-9456e8b34dd3', 'Status' is 'Active', and 'Account Nickname' is 'Scrub Test'. Navigation buttons '1' and '1-1/1' are at the bottom. A 'Help' button is at the bottom right.

4. Click on the three dots under the action tab.

The screenshot shows the Scrut Automation web application. On the left, there is a sidebar with the following navigation items:

- Dashboard
- Reports
- Compliance
- Risk
- Trust
- Audit
- People
- Product Updates
- Settings
- Integrations** (highlighted in black)

The main content area is titled "**← Amazon Web Services**". It features a sub-header "Amazon Web Services" with the AWS logo. Below it are two buttons: "Test" and "Scrut Monitor". A green button labeled "Connected" is visible in the top right corner of this section.

The main table header includes columns for "Integration ID ↑↓", "Status ↑↓", "Account Nickname ↑↓", "Entity ↑↓", and "Actions ↑↓". A single row is present in the table, showing "3a9fb1d16-04a6-4dc3-a26b-9456e8b34dd3" (Status: Active, Entity: Scrut Test), with a red box highlighting the "Actions" column. The table footer shows page number "1" and "1-1/1".

At the bottom left, there is a user profile icon for "JD John Doe". At the bottom right, there is a blue "Help" button with a question mark icon.

5. Apps in scope window opens.

Toggle on the switch next to the repositories for which you want to enable auto-collection of pull request evidence.

The screenshot shows the Scrut Automation interface for managing integrations. On the left sidebar, 'Integrations' is selected. The main content area is titled 'Amazon Web Services' and shows a 'Test' and 'Scrut Monitor' tab. A modal window titled 'Apps in Scope' is open, containing a search bar and a 'Mark All In' button. Below the modal, a table lists a single integration entry: '3a9fb16-04a6-4dc3-a26b-9456e8b34dd3' with status 'Active'. Navigation links like 'Dashboard', 'Reports', and 'Compliance' are also visible.

6. Once you have selected the repositories, click Save.

← Previous
Integrations in Scrut

Next →
Integrating Azure