



Cloud Protection

By WithSecure

Agentforce Extension

Manual for the Agentforce extension package

Contents

1 Agentforce extension overview.....	5
1.1 WithSecure Cloud Protection for Agentforce use cases.....	5
2 WithSecure Cloud Protection for Agentforce Deployment.....	7
2.1 Prerequisites.....	7
2.2 WithSecure Cloud Protection for Agentforce Installation using the AppExchange Marketplace.....	8
2.3 WithSecure Cloud Protection for Agentforce installation using Salesforce Portal.....	9
2.4 URL scanning setup.....	9
2.5 Configuring with the legacy agent builder.....	9
2.5.1 Agent Actions Configuration.....	10
2.5.2 Assigning the Topic to your Agent.....	10
2.5.3 Topic Configuration.....	10
2.5.4 Advanced Agent Topic Configuration.....	11
2.6 Configuring with Agent Script support.....	13
2.6.1 Switching to Script View and declaring internal state variables.....	13
2.6.2 Adding and calling the URL Scan action.....	14
2.6.3 Saving and testing.....	15
2.7 Permission set assignment.....	16
3 Agentforce Action Alerts notifications.....	17
3.1 Email notifications setup for the Agentforce Actions.....	17

Chapter 1 Agentforce extension overview

Salesforce is strategically evolving its platform by investing in Agentforce application development, leveraging cutting-edge technologies such as generative AI, Data Cloud, and large language models (LLMs). These innovations aim to enhance automation across core Salesforce products, including Service Cloud, Marketing Cloud, and Experience Cloud, by enabling intelligent agents to perform context-aware actions based on user interactions.

As part of this transformation, Agentforce Actions allow AI agents to respond to user prompts by invoking custom workflows. These workflows often involve making REST API calls to internal or external systems to retrieve URLs that are used to update or interact with Salesforce records. Given the dynamic nature of these interactions, it is critical to ensure that the retrieved URLs are safe, not harmful, and do not belong to disallowed categories.

To address this need, the WithSecure Cloud Protection for Agentforce introduces a protection layer that validates URLs before they are processed. This is especially important in real-time communication scenarios enabled by Agentforce Chat, a feature integrated with Salesforce's Service Cloud. Agentforce Chat facilitates live conversations with customers and stores transcript records that may be linked to Cases, Contacts, or Accounts.

In AI-driven conversations, users may inadvertently or maliciously provide URLs that attempt to manipulate sensitive data, such as modifying case records or injecting harmful content into transcripts. Without proper validation, these actions could compromise data integrity and pose security risks.

The WithSecure Cloud Protection for Agentforce ensures that every URL retrieved through Agentforce Actions is thoroughly checked, helping Salesforce Admins maintain a secure environment while enabling powerful automation through AI agents.

1.1 WithSecure Cloud Protection for Agentforce use cases

This chapter outlines key scenarios where the WithSecure Cloud Protection for Agentforce package enhances security and operational efficiency within Salesforce environments. By integrating real-time URL scanning into AI-powered workflows, the extension helps prevent malicious content from compromising internal data or customer interactions.

AI Assistance for Internal Operations

Problem:

The Agentforce agent helps employees work more efficiently by assisting with the creation, update, and modification of records. However, internal users can unknowingly paste URLs from external sources into the agent conversation. If those URLs are malicious, they can be acted on or passed to downstream actions before anyone reviews them.

Solution:

WithSecure Cloud Protection for Agentforce adds real-time URL scanning to the Agentforce agent. When a URL is flagged, the extension alerts the user and flags the harmful link in the conversation so the user does not pick it up, and stops the flagged content from being passed to downstream actions; it does not alter or redact the URL itself. URLs that are saved into Salesforce records are scanned separately by the URL protection in the main WithSecure Cloud Protection for Salesforce package.

The employee agent can be accessed from both Salesforce and Slack; the extension's URL Scan action also works with Agentforce for Slack.

Salesforce admins can configure the WithSecure Cloud Protection for Agentforce package to:

- Scan URLs in the agent conversation for harmful or disallowed content.
- Define a disallowed URL list.
- Receive alerts when threats are detected.
- Stop flagged content from being passed to downstream actions.

Service Agent Support

Problem:

Salesforce customers previously used tools like web chat, MIAW and live chat to support personalized customer interactions. Now, with the introduction of Agentforce, much of this interaction is automated by AI agents, and in some cases the employee agents and AI agents work together. These AI agents engage with guest users via Experience Cloud and external users embedded into third-party applications. However, they lack built-in content scanning, which increases the risk of malicious URLs being introduced into Salesforce through these interactions.

Solution:

WithSecure Cloud Protection for Agentforce enables real-time scanning of URLs during AI-powered service interactions. If the AI agent cannot resolve an issue and escalates it to a human agent, the extension acts as a safeguard by alerting the agent to avoid clicking on any potentially harmful or disallowed URLs that may have been shared with the AI agent.

This human agent handoff safeguard applies to any external communication — in particular, the Service Agent talking to customers over Messaging (in-app and web).

Administrators can configure the WithSecure Cloud Protection for Agentforce package to:

- Detect malicious URLs
- Define disallowed URL categories
- Receive alerts when threats are detected

This use case ensures secure customer communication and protects agents from interacting with unsafe content.

Chapter 2 WithSecure Cloud Protection for Agentforce Deployment

This section provides instructions for deployment and setting up the WithSecure Agent Action Extension package in your Salesforce organization.

Deploying the WithSecure Cloud Protection for Agentforce involves the following steps:

- Installing the WithSecure Cloud Protection for Agentforce
- Setting up the URL scanning in Agentforce Actions.

2.1 Prerequisites

Before deploying the WithSecure Cloud Protection for Agentforce package, ensure the following conditions are met.

Cloud Protection for Salesforce is installed

Important: Make sure that the Cloud Protection for Salesforce application **2.9.1 or above** is installed prior to installing the WithSecure Cloud Protection for Agentforce package. The WithSecure Cloud Protection for Agentforce extension depends on core components provided by the main application.

Note: If you need to upgrade the Cloud Protection for Salesforce application, please activate the picklist values before the update ([Activating picklist values before update](#)).

Note: If you have not installed the Cloud Protection for Salesforce you can download it from the AppExchange:

- [Global AppExchange](#)
- [Japanese AppExchange](#)

In order to install and configure the Cloud Protection for Salesforce use the steps available here: [Installing Cloud Protection for Salesforce application](#). You need admin permissions to the salesforce organization to install the application.

WithSecure Cloud Protection for Agentforce package is installed

Once the Cloud Protection for Salesforce application has been installed, you need to install the WithSecure Cloud Protection for Agentforce extension.

Note: There are two possible ways to install the new extension:

1. You can download the WithSecure Cloud Protection for Agentforce package directly from the [AI Agent Marketplace for Agentforce - AgentExchange](#)

2. You can download the WithSecure Cloud Protection for Agentforce directly from the Salesforce portal (for this second option please use the steps described in the next section of this chapter).

Attention: You need to have the Agentforce credits in your Salesforce organization to install the solution.

Agentforce Action Configuration Settings are enabled

Once the Cloud Protection for Salesforce application, version 2.9.1 or above, and the WithSecure Cloud Protection for Agentforce extension have been installed, you need to enable the following configuration settings:

- **Harmful URL Scanning**
- **Disallowed URL Scanning**

Note: In order to enable these settings please use the instructions available under the section: **URL scanning setup**.

Important: These settings will only take effect after the WithSecure Cloud Protection for Agentforce package has been installed. If the extension is not present, enabling these options in the Cloud Protection for Salesforce application will have no functional impact.

2.2 WithSecure Cloud Protection for Agentforce Installation using the AppExchange Marketplace

This section provides the instructions to install the WithSecure Cloud Protection for Agentforce package directly from the AppExchange Marketplace.

Follow these instructions to install the WithSecure Cloud Protection for Agentforce directly from the AppExchange marketplace.

1. Go to the [AI Agent Marketplace for Agentforce - AgentExchange](#).
2. Search for WithSecure Cloud Protection for Agentforce extension and open it..
3. Click **Get It Now**.

Important: Ensure that the Cloud Protection for Salesforce version 2.9.1 or above is installed in your Salesforce Organization. The WithSecure Cloud Protection for Agentforce package depends on core components provided by the Cloud Protection for Salesforce application.

Note: To install the Cloud Protection for Salesforce package please refer to the instructions available in [User Guide Manual](#) for Cloud Protection for Salesforce.

The WithSecure Cloud Protection for Agentforce package has been installed successfully.

2.3 WithSecure Cloud Protection for Agentforce installation using Salesforce Portal

This section provides the instructions to install the WithSecure Cloud Protection for Agentforce package directly from the Salesforce Portal.

Follow these instructions to install the WithSecure Cloud Protection for Agentforce package directly from the Salesforce portal.

1. Navigate to **Topics**.
2. Open **New** → **Add from AgentExchange**.
3. Select the WithSecure Cloud Protection for Agentforce package.
4. Install the package.

Please add the Agent Action directly from Add from **Asset Library**, if the extension package is already installed.

Important: Ensure that the Cloud Protection for Salesforce application version 2.9.1 or above is installed in your Salesforce Organization. To install the Cloud Protection for Salesforce application please refer to the instructions available in the following: [User Guide Manual](#). The WithSecure Cloud Protection for Agentforce package depends on core components provided by the Cloud Protection for Salesforce application.

The WithSecure Cloud Protection for Agentforce package has been installed successfully.

2.4 URL scanning setup

After you have installed the WithSecure Cloud Protection for Agentforce package, you need to setup the URL scanning for the Agentforce Actions.

Follow these instructions to setup the URL scanning for the Agentforce Actions.

1. Go to **Administration** and open **URL Protection**.
2. Enable the **Scan URLs** toggle under the **General** section
3. Enable the following settings under the **Settings**:
 - Include URLs from agent actions in harmful URL scanning.
 - Include URLs from agent actions in disallowed URL scanning.

Important: These settings will only take effect after the WithSecure Cloud Protection for Agentforce package has been installed. If the extension is not present, enabling these options in the Cloud Protection for Salesforce application will have no functional impact.

2.5 Configuring with the legacy agent builder

Salesforce provides two ways to configure the agent. Follow the steps in this section if you are using the legacy agent builder experience.

Use the steps below to configure the agent action, assign the topic, and configure the topic in the legacy agent builder. If you are using the new Agent Builder console (Agent Studio) with Agent Script support, see [2.6 Configuring with Agent Script support](#) instead.

2.5.1 Agent Actions Configuration

This section provides the information to configure the Agent Actions to use with the Service Agent.

Follow these instructions to configure the Agent Actions.

1. Navigate to **Setup** and search for the **Agentforce Studio**.
2. Under the **Agentforce Studio** open the **Agentforce Agents**.
3. Select the type of the **Agent** needed to be secured against malicious content.
4. Click the downward arrow to choose the option **Open in Builder**.
5. Deactivate the agent until the settings are adjusted, as recommended by Salesforce ([Activate or Deactivate Your Agent](#)).

2.5.2 Assigning the Topic to your Agent

You need to select the topics assigned to your Agent.

Remember: To make changes, deactivate the agent until the settings are adjusted, as recommended by Salesforce. ([Activate or Deactivate Your Agent](#))

Follow the instructions below to select the topic assigned to your Agent.

1. Go to **Topics**.
2. Create the **Topic Label** according to your requirement.
3. Select the **Topic Action**.
For reference we have created the **Customer Experience Support** topic label.
4. Go to **This topic's Actions**.
5. Click on **New** and select **Add from Asset Library**.
6. Select the **Scan URLs** topic action.

Note: If the WithSecure Cloud Protection for Agentforce package is not installed then you can install the package by selecting **Add from AgentExchange**. The **Scan Url** action is visible in the asset library once the extension package is installed.

7. Go to **Topics** section.
8. Select the **Topic Label: Customer Experience Support**.
9. Choose the **Scan URLs** action under **This Topic's Actions**.
10. Click **Finish**.

2.5.3 Topic Configuration

This section outlines the steps for adding the instructions to the Topic.

Follow the steps below to include the instructions to the **Topic** .

1. Go to **Topic** → **Topic Details**.
2. Add the instructions under **Topic Configuration**.

Note: The instructions should be customized and added according to the organization's requirements.

Tip: Below you can find the **Example of the instruction for scanning malicious URLs**.

1. Always pass user input to the Apex action `Scan_Url`s. This action must be executed **with the highest priority** and **before any other action**. The response should be displayed in the user chat window using the **"message" component** with **rich text output**. If output is empty, do not mention anything related to URL scanning to the user and proceed to next actions.

Important:

The output must clearly display **harmful URLs** and **disallowed URLs** in **two separate sections**.

This is a **high-priority task** and must be enforced consistently.

2. No other actions should proceed until `Scan_Url`s has been executed and its response shown to the user.

3. Save the changes and Activate the Agent.

2.5.4 Advanced Agent Topic Configuration

This section outlines the steps for the advanced configuration of the Agent Topic in the legacy agent builder.

The advanced Agent Topic configuration includes the following components:

- Variables Setup
- Filters Configuration
- The output variables mapping

Variables Setup

Variables are key components that control and enhance agent behavior by providing deterministic logic and secure data handling for topic and action selection.

Follow these instructions to setup the Variables.

1. Navigate to the **Context** section and locate the **Variables**.
2. Click on **New Variable**.
3. Enter:
 - **API Name**
 - **Description**
 - **Data type as: Boolean**

Note: You will find two examples of Boolean variables under **Custom Variables**:

- Has Harmful Content with **API name:** Has_Harmful_Content and **Data type** as: Boolean.
- Has Disallowed Content with **API name:** Has_Disallowed_Content and **Data type** as: Boolean.

The variable **Has Harmful Content** protects against harmful URLs, while the variable **Has Disallowed Content** safeguards against disallowed URL categories.

4. Select **Has Harmful Content** and **Has Disallowed Content** or the variable name you have created.

Note: Please don't select **Allow value to be set by API** or **Allow LLM to use value**. Selecting **Allow LLM to use value** enables AI to use the value for its decisions and output.

Filters Configuration

The primary purpose of the filters is to stop an action from executing with the specified variable when harmful or disallowed content is detected.

Follow these instructions to setup Filters.

1. Navigate to **Filters** in **Context** section.
2. Click **New**.
3. Enter the **Name** and provide **Filter Conditions**.
4. In the **Resource option**, select the variable created in the previous steps.
5. Add the **Operator** and **Value**.

Example:

1. Filter 1: Is Safe to Proceed (Harmful)

- **Resource** → **Has Harmful**
- **Operator** → **Does Not Equal**
- **Value** → **True**

This will scan the URLs interacted with in Agent for harmful content and take action based on the settings in the Cloud Protection for Salesforce.

2. Filter 2: Is Safe to Proceed (Disallowed)

- **Resource** → **Has Disallowed**
- **Operator** → **Does Not Equal**
- **Value** → **True**

This will scan the disallowed URLs interacted with in the Agent for disallowed content and take action based on the settings in the Cloud Protection for Salesforce application.

Mapping the output variables

You need to map the correct output variables from the **Scan URLs** action.

1. Map the correct output variables from the **Scan URLs** action.
 - a) containsDisallowedUrl -> **Has disallowed**.
 - b) containsHarmfulUrl -> **Has Harmful**.
 - c) message → **Output Rendering - Rich Text**.
2. Use the filters to restrict any actions when unsafe or disallowed content is detected.

Note: You can create your own filters using the boolean values: **Has Harmful Content** and **Has Disallowed Content**. The filters can be applied to any other action other than **Scan URLs**.

Attention: You shouldn't add these filters to the **Scan URLs** Agent Action.

2.6 Configuring with Agent Script support

Follow the steps in this section if you are using the new Agent Builder console (Agent Studio) with Agent Script support.

This section shows how to wire the WithSecure Cloud Protection for Agentforce URL Scan action (`wsaa__FS_AgentURLScanner`) into an Agentforce agent authored with the YAML-based Agent Builder. If you are using the legacy agent builder experience, see [2.5 Configuring with the legacy agent builder](#) instead.

Prerequisites:

- WithSecure Cloud Protection for Agentforce package installed in the target org (includes the `wsaa__Scan_Urls` invocable action backed by apex `://wsaa__FS_AgentURLScanner`).
- An Agentforce agent authored via the Agent Builder YAML format (Agent Definition).

Where to add the URL scan

Add the URL scan to the **Agent Router**, in the `before_reasoning` hook. It runs once, deterministically, on every message before the agent does anything with it.

- **Why the Agent Router, and not each subtopic?** Every message passes through the router, so one scan covers all of them. A scan placed in a subtopic only covers messages routed there, and a phishing or malicious link can slip through any subtopic you miss — including ones added later.
- **Why before_reasoning, and not an action the agent chooses to run?** Anything the LLM decides to call can drift: it may skip the scan or run it at the wrong moment. `before_reasoning` is deterministic and runs before the LLM is called, so a flagged message is stopped before it reaches a subagent or the agent acts on it. It is also faster and cheaper, because the check runs on Salesforce resources before the message reaches the LLM.

2.6.1 Switching to Script View and declaring internal state variables

Switch to Script View, then add mutable, internal-only variables to track scan results, session state, and the messaging session id.

1. Switch from the **UI** view to **Script View** by clicking the `</>` button.

Note: Use Script View for these steps — many of the features used here are not accessible from the UI view.

2. Add the following mutable, internal-only variables under the top-level `variables` block.

Search (Cmd/Ctrl+F) for `variables` and use the **top-level** block — the one that is not indented and is not inside `start_agent` `agent_router` or any subagent. Go to the last variable in that

block, place the cursor at the end of its last line (usually its `visibility:` or `description:` line), and paste the variables below. Indent each variable name one level under `variables:` (4 spaces) and its `description:` and `visibility:` lines one level deeper (8 spaces), matching the variables already there.

These three fields track whether the last user message contained any unsafe URLs:

```
Has_Disallowed_Content: mutable boolean = False
  description: "Stores whether the message contains a disallowed URL"
  visibility: "Internal"
Has_Harmful_Content: mutable boolean = False
  description: "Stores whether the message contains a harmful URL"
  visibility: "Internal"
Scan_Message: mutable string = ""
  description: "Rich-text result returned by the URL scan, shown to the user when content is flagged."
  visibility: "Internal"
```

This field tracks whether any unsafe URL was posted during the current session:

```
Has_Detected_Unsafe_In_Session: mutable boolean = False
  description: "Permanently latched to True once unsafe content is detected in any turn. Never reverted to False."
  visibility: "Internal"
```

This field passes the messaging session id, when available, to the `Scan_Urls` action (`RoutableId` = `@MessagingSession.Id`):

```
EffectiveRoutableId: mutable string = "None"
  description: "Value of the RoutableId (@MessagingSession.Id), used for the Scan URLs action; 'None' when RoutableId is null."
  visibility: "Internal"
```

2.6.2 Adding and calling the URL Scan action

Add the `Scan_Urls` action, call it from the router's `before_reasoning` hook, and guard the router actions so flagged content never reaches a subagent.

1. Add the `Scan_Urls` action from the Asset Library.

Map the action's input and output variables as follows.

Inputs:

Input	Type	Source
<code>textToScan</code>	string	<code>@system_variables.user_input</code> (the raw text the end user just sent)
<code>messagingSessionId</code>	string	<code>@variables.EffectiveRoutableId</code>

Outputs:

Output	Type	Purpose
<code>containsDisallowedUrl</code>	boolean	True if a disallowed URL was found.
<code>containsHarmfulUrl</code>	boolean	True if a harmful or malicious URL was found.
<code>message</code>	rich text	Human-readable explanation to show the user when content is flagged.

2. Call the action from the router's `before_reasoning` block.

This block runs deterministically, before the LLM is called, on every message. Search (Cmd/Ctrl +F) for `start_agent agent_router :`, then for `reasoning :` inside it, and paste this block directly above `reasoning :`, at the same indentation. It invokes `Scan_Urls`, maps its outputs into the internal variables, and latches `Has_Detected_Unsafe_In_Session` when the message is flagged.

```
before_reasoning:
  if @variables.RoutableId != None: # Set the RoutableId (@MessagingSession.Id) if it exists.
    set @variables.EffectiveRoutableId = @variables.RoutableId
  run @actions.Scan_Urls
    with messagingSessionId = @variables.EffectiveRoutableId
    with textToScan = @system_variables.user_input
    set @variables.Has_Disallowed_Content = @outputs.containsDisallowedUrl
    set @variables.Has_Harmful_Content = @outputs.containsHarmfulUrl
    set @variables.Scan_Message = @outputs.message
  if @variables.Has_Disallowed_Content == True or @variables.Has_Harmful_Content == True:
    set @variables.Has_Detected_Unsafe_In_Session = True
```

Save and check that the script shows no errors.

3. Guard the router actions so a flagged message cannot reach a subagent.

The `before_reasoning` block only detects unsafe URLs and stores the result; it does not stop anything by itself. Add an `available when` condition to each action in the router's `reasoning :` → `actions :` list. While the condition is false, the action is hidden from the LLM, so a flagged message has nowhere to go. Choose one of the following guarding options and use it on every action:

- Gate on the latest message scan result — stops only the message that contains an unsafe URL from reaching a subagent; the next clean message works normally:

```
available when not @variables.Has_Harmful_Content and not @variables.Has_Disallowed_Content
```

- Gate on any unsafe URL posted during the session — stops every message after the first unsafe URL, for the rest of the session:

```
available when not @variables.Has_Detected_Unsafe_In_Session
```

Example actions with the condition added:

```
actions:
  go_to_GeneralFAQ: @utils.transition to @subagent.GeneralFAQ
    available when not @variables.Has_Harmful_Content and not @variables.Has_Disallowed_Content
  go_to_search_contacts: @utils.transition to @subagent.search_contacts
    available when not @variables.Has_Harmful_Content and not @variables.Has_Disallowed_Content
  go_to_off_topic: @utils.transition to @subagent.off_topic
    available when not @variables.Has_Harmful_Content and not @variables.Has_Disallowed_Content
  go_to_ambiguous_question: @utils.transition to @subagent.ambiguous_question
    available when not @variables.Has_Harmful_Content and not @variables.Has_Disallowed_Content
```

Note: For a stricter response, you can use the same flags to end the session. Show `Scan_Message` to the user before the session ends, so they know why.

2.6.3 Saving and testing

Save your changes and verify the URL Scan action in the simulator.

Save the change and refresh the simulator before testing.

2.7 Permission set assignment

This section provides the instructions to assign permissions set for the Agent user. Agentforce permission set is mandatory for the URL scans invocation.

To assign permissions set for the Agent user please follow the instructions below.

1. Go to **Settings** section in **Agent Builder** and identify the **Agent User** i.e. **Agentforce Service Agent (ASA)**.
2. Assign the following permissions sets:
 - For **Internal users**:
 - if the **WithSecure Cloud Protection User** permission set is assigned, assign only the **Cloud Protection for Salesforce Agent User - Extension**.
 - if the **WithSecure Cloud Protection User** permission set is not assigned, assign both permissions sets: the **Cloud Protection for Salesforce Agent User** and the **Cloud Protection for Salesforce Agent User - Extension**.
 - For the **Agentforce Service Agents (ASA)** with the Einstein Agent User Profile:
 - assign both permission sets: the **Cloud Protection for Salesforce Agent User** and **Cloud Protection for Salesforce Agent User - Extension**.

Important: The following both permission sets must be assigned to internal and service agent users to ensure necessary content protection.

- Cloud Protection for Salesforce Agent User
- Cloud Protection for Salesforce Agent User – Extension

Once the above settings are completed:

1. Activate the Agent ([Activate or Deactivate Your Agent](#))
2. Refresh the **Conversation Preview**.
3. Test the Agent according to the configured settings in the Cloud Protection for Salesforce application.

Chapter 3 Agentforce Action Alerts notifications

The **WithSecure Agent Action** extension package includes a robust alerting and event generation system designed to enhance visibility and security during AI-powered agent interactions. To ensure timely awareness of security threats detected during the AI agent interactions, the **WithSecure Agent Action Extension Package** includes configurable email notifications for harmful and disallowed URLs.

Key Capabilities:

1. Event Generation for All URLs

Every URL posted in the agent chat is logged as an event. This ensures full traceability of user interactions.

2. Alerting for Malicious or Disallowed URLs

If a URL is identified as malicious or belongs to a disallowed category, an alert is automatically generated. This helps to respond quickly to potential threats.

3. System Alerts for Configuration and Connectivity Issues

Alerts are also triggered when:

- Agentforce scan settings are modified
- The extension package becomes disconnected
- Errors are detected in the extension package

4. Detailed URL Event Metadata

Each URL event includes the following information:

- Date/Time
- Verdict (e.g., Safe, Malicious)
- Action taken
- Reason for the verdict
- URL
- Direction (Inbound/Outbound)
- Location (e.g., Chat, Transcript)
- User
- IP Address

3.1 Email notifications setup for the Agentforce Actions

This section provides the instruction to configure the email notification for the **Agentforce Action**.

Follow the instructions below to setup the email notifications for the **Agentforce Action URL scanning** and for **Agentforce Action Disallowed URL**

1. Go to **Administration** and open the **URL Protection**.

2. Under the **Notifications** section turn on the toggle for:

- **Send a security alert when a harmful url from agent action is detected**
- and
- **Send a security alert when a disallowed url from agent action is detected.**

The email notifications for the Agentforce Actions have been successfully set.