Azure SRE Agent (preview) Documentation

Azure SRE Agent overview

Site Reliability Engineering (SRE) focuses on creating reliable, scalable systems through automation and proactive management. Azure SRE Agent brings these principles to your Azure hosted applications by providing an AI-powered tool that helps sustain production cloud environments. SRE Agent helps you respond to incidents quickly and effectively, alleviating the toil of manually managing production environments. The agent uses the reasoning capabilities of large language models (LLMs) to identify the logs and metrics necessary for rapid root cause analysis and issue mitigation. Azure SRE Agent brings you better service uptime and reduced operational costs.

Agents have access to every resource inside the resource groups associated with the agent. Therefore, agents:

- Continuously evaluate resource activity, and monitor active resources

- Send proactive notifications about unhealthy or unstable apps

Azure SRE Agent also integrates with Azure Monitor Alerts and PagerDuty to support advanced notification solutions.

By using an SRE Agent, you consent the product-specific Supplemental Terms of Use for Microsoft Azure Previews.

Key features

Azure SRE Agent offers several key features that enhance the reliability and performance of your Azure resources:

- **Welcome thread**: When you first create your agent, a new thread is created which provides initial analysis of your services. The environment analysis creates a snapshot of all the resources managed by the agent. Additionally, the agent generates a list of applications found in the managed resource groups.

- **Daily threads**: Each day, the agent creates a resource report which summarizes the state and status of the services in your managed resource groups.

- **Tooling**: Querying and operations support via Azure CLI and Kubectl.

- **Data sources**: Access to Azure Resource Manager APIs and Azure Monitor metrics data sources.

- **Incident management**: Diagnose incidents by chatting with the agent directly or by connecting an incident management platform to the agent. Automatically respond to Azure Monitor alerts or PagerDuty incidents with initial analysis.

- **Proactive monitoring**: Continuous 24x7 resource monitoring with real-time alerts for potential issues.

- **Automated mitigation:** Automatic detection and mitigation of common issues, reducing downtime and improving resource health. While agents attempt to work on your behalf, all automation requires your approval.

- **Infrastructure best practices:** Identify and remediate resources not following security best practices and help updates.

- **Accelerates root cause analysis:** Diagnose root causes of app issues by analyzing metrics and logs and suggest mitigations.

- **Resource visualization**: Comprehensive views of your resource dependencies and health status.

- **Mitigation support**: SRE Agent can fix application configuration and dependent services. For code issues, the agent provides stack traces and can create GitHub issues to help resolve issues. The following describes service-specific features of the agent:

    - *Azure App Service*: Roll back deployment, scale resources up/down, application restarts.

    - *Azure Container Apps*: Roll back deployment, scale resources up/down, and application restarts.

    - *Azure Kubernetes Service*: Restart pods/deployments, roll back deployments to previous revisions, scale resources up/down, and patch resource definitions.

Reports

An SRE Agent works to proactively monitor and maintain your Azure services. Each day your agent creates daily resource reports which provide insights into the health and status of your applications.

Reports include:

- **Incident summary:** Generates information about incidents raised by the SRE Agent on the previous day. Categories include: active, mitigated, or resolved.

- **Application group performance and health:** Key metrics for each application group to assess system stability and performance. Metrics include: availability, CPU usage, and memory usage.

- **Action summary:** Summaries of important details and insights relevant to the health and maintenance of your Azure resources.

Scenarios

Scenario	**Possible cause**	**Agent mitigation**
Application down	▪ **Application code issues**: Bugs or errors in the application code can lead to crashes or unresponsiveness. ▪ **Bad deployment**: Incorrect configurations or failed deployments can cause the application to go down. ▪ **High CPU/memory/thread issues**: Resource exhaustion due to high CPU, memory, or thread usage can affect application performance.	The SRE Agent can detect these issues and provide actionable insights or fixes. For example, it can identify a decrease in web app availability that coincides with a recent slot swap and recommend swapping back slots as the first step of mitigation.
Container image pull failures	▪ **Image availability**: The requested image might not be available or could be missing. ▪ **Network connectivity**: Network issues can disrupt the connection to the container app. ▪ **Registry connectivity issues**: Problems with connecting to the container	The SRE Agent can detect container image pull failures and provide detailed diagnostics. It can recommend solutions such as rolling back to the last known healthy revision and updating the image reference.

Scenario	Possible cause	Agent mitigation
	registry can prevent image pulls.	

An agent can provide detailed information about different aspects of your apps and resources. The following examples demonstrate the types of questions you could pose to your agent:

- What can you assist me with?
- Why isn't my application working?
- What services is my resource connected to?
- Can you provide best practices for my resource?
- What's the CPU and memory utilization of my app?

Further, here are some prompts you can use to help you interact with your agent:

- Which apps have Dapr enabled?
- List replicas for my container app
- Which apps have diagnostic logging turned on?
- Give me an individual heatmap for each storage account.
- Which revision of my container app is currently active?
- What are some best practices that my app should follow?
- What is the ingress configuration for my container app?
- Are there any staging slots configured for this web app?
- What container images are used by each of my Container Apps?
- List all resource groups that you’re managing across all subscriptions.
- Draw heatmap of storage latencies over the last 14 days for storage accounts.
- Show me a visualization of response times for Container Apps for last week.
- List [Container Apps/Web Apps/etc.] that you’re managing across all subscriptions.
- Visualize split of Container Apps vs Web Apps vs AKS clusters managed across all subscriptions as a pie chart.

Create and use the Azure SRE Agent

Azure SRE Agent helps you maintain the health and performance of your Azure resources through AI-powered monitoring and assistance. Agents continuously watch your resources for issues, provide troubleshooting help, and suggest remediation steps available through a natural language chat interface. To ensure accuracy and control, any agent action taken on your behalf requires your approval.

This article demonstrates how to create an SRE Agent, connect it to your resources to maintain optimal application performance.

Run modes

Azure SRE Agent operates in one of three different modes. Your agent behaves differently, depending on the mode type you select.

The three different types of modes are:

- **Read-only**: The read-only mode puts your agent in an observation mode. The agent has access to your inspect and report on your apps and can advise you on what actions to take. In this mode, the agent only has *reader* access to most services. In limited instances, the agent is granted *contributor* access to services solely to access configuration data.

- **Review**: As the agent operates in review mode, the agent can make changes to your apps and services on your behalf, but doesn't take action unless you give express approval. In this mode, the agent has *reader* or *contributor* access to services.

- **Autonomous**: Autonomous mode gives the agent full ability to work on your behalf without the need to request approval to proceed. In this mode, the agent has *reader* or *contributor* access to services.

Create an agent

Create an agent by associating resource groups you want to monitor to the agent.

Prerequisites

You need to grant your agent the correct permissions and access to the right namespace.

- **Security context**: Before you can create a new agent, make sure your user account has the `Microsoft.Authorization/roleAssignments/write` permissions using either Role Based Access Control Administrator or User Access Administrator.

- **Namespace**: Using the cloud shell in the Azure portal, run the following command:

  ```
  az provider register --namespace "Microsoft.App"
  ```

- **Associate your allow list subscription ID**: Make sure your Azure CLI session is set to the subscription ID on the preview allow list. If you need to set the CLI context to your subscription ID, use the following command:

  ```
  az account set --subscription "<SUBSCRIPTION_ID>"
  ```

Create

To create an SRE Agent, follow these steps:

1. Go to the SRE Agent Azure portal and search for and select **Azure SRE Agent**.

2. Select **Create**.

3. Enter the following values in the *Create agent* window:

   During this step, you create a new resource group specifically for your agent which is independent of the resource group used for your application.

   In the *Project details* section, enter the following values:

Property	**Value**
Subscription	Select your Azure subscription.
Resource group	Select an existing resource group or to create a new one, enter a name.

   In the *Agent details* section, enter the following values:

Property	**Value**
Agent name	Enter a name for your agent.
Region	Select **Sweden Central**.

Property	Value
	During preview, Azure SRE Agent is only available in the *Sweden Central* region, but the agent can monitor resources in any Azure region.
Run mode	Select **Review*. When in *review mode*, the agent works on your behalf only with your approval.

4. Select **Choose resource groups**.

5. In the *Choose resource groups to monitor* window, search for the resource group you want to monitor.

   Avoid selecting the resource group link.

   To select a resource group, select the checkbox next to the resource group.

6. Scroll to the bottom of the dialog window and select **Save**.

7. Select **Create**.

   Once you begin the create process, a page with the message *Deployment is in progress* is displayed.

Chat with your agent

Your agent has access to any resource inside the resource group associated with the agent. Use the chat feature to help you inquire about and resolve issues related to your resources.

1. Go to the Azure portal, search for and select **Azure SRE Agent**.

2. Locate your agent in the list and select the agent name.

Once the chat window loads, you can begin asking your agent questions. Here's a series of questions that can help you get started:

- What can you help me with?

- What subscriptions/resource groups/resources are you managing?
- What alerts should I set up for `<RESOURCE_NAME>`?
- Show me visualization of `2xx` requests vs HTTP errors for my web apps across all subscriptions

If you have a specific problem in mind, you could ask questions like:

- Why is `<RESOURCE_NAME>` slow?
- Why is `<RESOURCE_NAME>` not working?
- Can you investigate `<RESOURCE_NAME>`?
- Can you get me the `<METRIC>` of `<RESOURCE_NAME>`?

Update managed resource groups

You can change the list of resource groups managed by your agent at any time. To change the list of managed groups, go to your agent in the Azure portal and select the **Settings** tab and then **Managed resource groups**.

> **Note:** Removing resource groups from the list does not remove or otherwise adversely affect resource groups.

Incident management

You can diagnose incidents in Azure App Service, Azure Container Apps, Azure Function, Azure Kubernetes Service and Azure Database for PostgreSQL by chatting with the agent directly or by connecting an incident management platform.

By default SRE Agent connects to Azure Monitor, but you can also connect it to PagerDuty.

PagerDuty integration

To set up the SRE Agent with PagerDuty, you need a PagerDuty API key.

1. In your SRE Agent resource, go to the *Settings* tab and select **Incident Management**.
2. From the *Incident platform* dropdown, select **PagerDuty**.
3. Enter your API key.
4. Select **Save**.