



AWS Diagnostic Tools API Reference

# AWS Diagnostic Tools



**API Version 2023-01-01**

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# **AWS Diagnostic Tools: AWS Diagnostic Tools API Reference**

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

---

# Table of Contents

<b>Welcome</b> .....	<b>1</b>
<b>Actions</b> .....	<b>2</b>
GetExecution .....	3
Request Syntax .....	3
Request Parameters .....	3
Response Syntax .....	3
Response Elements .....	4
Errors .....	4
Examples .....	5
See Also .....	6
GetExecutionOutput .....	7
Request Syntax .....	7
Request Parameters .....	7
Response Syntax .....	7
Response Elements .....	7
Errors .....	8
Examples .....	9
See Also .....	10
GetTool .....	11
Request Syntax .....	11
Request Parameters .....	11
Response Syntax .....	11
Response Elements .....	12
Errors .....	12
Examples .....	13
See Also .....	14
ListExecutions .....	15
Request Syntax .....	15
Request Parameters .....	15
Response Syntax .....	15
Response Elements .....	16
Errors .....	16
Examples .....	17
See Also .....	19

---

ListTagsForResource .....	20
Request Syntax .....	20
Request Parameters .....	20
Response Syntax .....	20
Response Elements .....	20
Errors .....	21
Examples .....	22
See Also .....	23
ListTools .....	24
Request Syntax .....	24
Request Parameters .....	24
Response Syntax .....	24
Response Elements .....	25
Errors .....	25
Examples .....	26
See Also .....	27
StartExecution .....	29
Request Syntax .....	29
Request Parameters .....	29
Response Syntax .....	31
Response Elements .....	31
Errors .....	32
Examples .....	32
See Also .....	34
TagResource .....	35
Request Syntax .....	35
Request Parameters .....	35
Response Elements .....	36
Errors .....	36
Examples .....	36
See Also .....	37
UntagResource .....	39
Request Syntax .....	39
Request Parameters .....	39
Response Elements .....	40
Errors .....	40

---

Examples .....	40
See Also .....	41
<b>Data Types .....</b>	<b>43</b>
Execution .....	44
Contents .....	44
See Also .....	46
ExecutionSummary .....	47
Contents .....	47
See Also .....	49
Tag .....	50
Contents .....	50
See Also .....	50
Tool .....	51
Contents .....	51
See Also .....	52
ToolSummary .....	53
Contents .....	53
See Also .....	54
ToolVersion .....	55
Contents .....	55
See Also .....	56
ValidationExceptionField .....	57
Contents .....	57
See Also .....	57
<b>Common Parameters .....</b>	<b>58</b>
<b>Common Errors .....</b>	<b>61</b>

# Welcome

AWS Diagnostic Tools provides an HTTP endpoint to self diagnose common troubleshooting issues with AWS Services.

The AWS Diagnostic Tools includes several diagnostic tools you can run to diagnose various AWS services. The service includes a set of APIs designed to streamline and manage diagnostic processes. These APIs enable a range of actions: `GetTool` and `ListTools` provide information about the diagnostic tools available within the service. `StartExecution` initiates a diagnostic process, termed an `Execution`. Post-execution, `GetExecution` and `GetExecutionOutput` allow users to retrieve details and results of these `Executions`, respectively. For organizational efficiency, `Executions` can be tagged or untagged using `TagResource` and `UntagResource`. Additionally, `ListExecutions` offers a view of all `Executions`, and `ListTagsForResource` aids in viewing tags associated with a specific resource. Altogether, these APIs facilitate robust and efficient diagnostic operations within the DT service.

This document was last published on March 13, 2025.

# Actions

The following actions are supported:

- [GetExecution](#)
- [GetExecutionOutput](#)
- [GetTool](#)
- [ListExecutions](#)
- [ListTagsForResource](#)
- [ListTools](#)
- [StartExecution](#)
- [TagResource](#)
- [UntagResource](#)

# GetExecution

Retrieve an execution by its id.

## Request Syntax

```
{
  "identifier": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

The unique identifier for an existing troubleshooting execution to examine. The execution ID is returned by StartExecution

Type: String

Length Constraints: Minimum length of 12. Maximum length of 2048.

Required: Yes

## Response Syntax

```
{
  "execution": {
    "creationTime": number,
    "executionId": "string",
    "parameters": "string",
    "requesterArn": "string",
    "requesterId": "string",
    "requestState": "string",
    "roleArn": "string",
    "status": "string",
    "storageRegion": "string",
  }
}
```



```
"targetRegions": [ "string" ],  
"toolId": "string",  
"toolVersionId": "string"  
}  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### execution

Execution Information Metadata

Type: [Execution](#) object

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerError**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **ResourceNotFoundException**

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

## ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `GetExecution`.

### Example

#### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.GetExecution
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "executionId": "e-aaaaaaaa"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "execution": {
```

```
    "creationTime": 1700601,  
    "executionId": "e-aaaaaaaa",  
    "requestState": "SUBMITTED",  
    "requesterArn": "requesterArn",  
    "requesterId": "user-id",  
    "roleArn": "<sample role>",  
    "status": "CREATED",  
    "storageRegion": "us-east-2",  
    "tags": [],  
    "targetRegions": [  
        "us-east-1"  
    ],  
    "toolId": "EC2SystemsManager",  
    "toolVersionId": "1.0.0"  
  }  
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetExecutionOutput

Retrieve the output for an Execution using its ID.

## Request Syntax

```
{  
  "identifier": "string"  
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

The unique identifier for an existing execution. The execution ID is returned by StartExecution

Type: String

Length Constraints: Minimum length of 12. Maximum length of 2048.

Required: Yes

## Response Syntax

```
{  
  "executionOutput": "string"  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### executionOutput

Execution Output Object

Type: String

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **ConflictException**

Updating or deleting a resource can cause an inconsistent state.

HTTP Status Code: 400

### **InternalServerErrorException**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **OutputTooLargeException**

Request failed due to output content being too large.

HTTP Status Code: 400

### **ResourceInFailedStateException**

Request failed due to the execution failing.

HTTP Status Code: 400

### **ResourceNotFoundException**

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

## ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `GetExecutionOutput`.

### Example

#### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.GetExecutionOutput
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "executionId": "e-aaaaaaaa"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "executionOutput": "<output in String format>"
}
```

```
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetTool

Retrieve a tool (optionally a specific version) by its id and versionId.

## Request Syntax

```
{
  "identifier": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

Version id for tool which your want information

Type: String

Length Constraints: Minimum length of 8. Maximum length of 2048.

Required: Yes

## Response Syntax

```
{
  "tool": {
    "id": "string",
    "latestVersionId": "string",
    "name": "string",
    "versions": [
      {
        "description": "string",
        "labels": [ "string" ],
        "parametersSchema": "string",
        "usedFor": "string",
        "versionId": "string"
      }
    ]
  }
}
```



```
    }  
  ]  
}  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### tool

Information about a Diagnostic Tool.

Type: [Tool](#) object

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerError**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **ResourceNotFoundException**

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

## ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `GetTool`.

### Example

#### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.GetTool
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "identifier": "RDSClustersWithMetrics"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "tool": {
    "id": "EC2SystemsManager",
    "latestVersionId": "1.0.1",
```

```
    "name": "EC2 Systems Manager",
    "versions": [
      {
        "components": "",
        "description": "Lists the SSM Managed Instances, Activations,
Maintenance Windows, Automation Executions, Patch Baselines, Patch Groups, Parameters
and Documents",
        "parameters": "<Parameter Info>",
        "usedFor": "Investigating Systems Manager details (Managed Instances,
Activations, Patch Groups, and more). on the account.",
        "versionId": "1.0.1"
      }
    ]
  }
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListExecutions

List all of the executions available for the current user.

## Request Syntax

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### maxResults

The maximum number of items to return for this call. The call also returns a token that you can specify in a subsequent call to get the next set of results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

### nextToken

The token for the next set of items to return. (You received this token from a previous call.)

Type: String

Pattern: `[A-Za-z0-9\-\_]*={0,3}`

Required: No

## Response Syntax

```
{
  "executionList": [
    {
```

```
    "creationTime": number,
    "executionId": "string",
    "requesterArn": "string",
    "requesterId": "string",
    "requestState": "string",
    "roleArn": "string",
    "status": "string",
    "storageRegion": "string",
    "targetRegions": [ "string " ],
    "toolId": "string",
    "toolVersionId": "string"
  }
],
"nextToken": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### executionList

Array of Execution objects

Type: Array of [ExecutionSummary](#) objects

### nextToken

The token to use when requesting the next set of items. If there are no additional items to return, the string is empty.

Type: String

Pattern: `[A-Za-z0-9\-\_]*={0,3}`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerErrorException**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### **ValidationException**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## **Examples**

This example illustrates one usage of `ListExecutions`.

### **Example**

#### **Using AWS JSON protocol (default)**

#### **Sample Request**

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.ListExecutions
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "maxResults": 2
}
```

## Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "nextToken": "nextTokenToRetrieveNextPage",
  "executionList": [
    {
      "creationTime": 1699927.553,
      "executionId": "e-aaaaaaaa",
      "requestState": "SUBMITTED",
      "requesterArn": "<requester iam arn>",
      "requesterId": "<requester iam arn>",
      "roleArn": "<pass role>",
      "status": "SUCCESSFUL",
      "storageRegion": "us-east-2",
      "targetRegions": [
        "us-east-1"
      ],
      "toolId": "EC2SystemsManager",
      "toolVersionId": "1.0.0"
    },
    {
      "creationTime": 1699927.553,
      "executionId": "e-bbbbbbbb",
      "requestState": "SUBMITTED",
      "requesterArn": "<requester iam arn>",
      "requesterId": "<requester iam arn>",
      "roleArn": "<pass role>",
      "status": "SUCCESSFUL",
      "storageRegion": "us-east-1",
      "targetRegions": [
        "us-west-2"
      ],
      "toolId": "EC2SystemsManager",
      "toolVersionId": "1.0.1"
    }
  ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# ListTagsForResource

Returns a list of the tags assigned to the specified resource.

## Request Syntax

```
{  
  "identifier": "string"  
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

The resource identifier for which you want to see a list of tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

## Response Syntax

```
{  
  "tags": [  
    {  
      "key": "string",  
      "value": "string"  
    }  
  ]  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## tags

A list of tags.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerError**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **ResourceNotFoundException**

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### **TooManyTagsException**

Request exceeded maximum number of tag allowed..

HTTP Status Code: 400

### **ValidationException**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `ListTagsForResource`.

### Example

#### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.ListTagsForResource
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "identifier": "e-aaaaaaaaaa"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "tags": [
    {
      "key": "department",
      "value": "support"
    }
  ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListTools

List all of the available tools.

## Request Syntax

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### maxResults

The maximum number of items to return for this call. The call also returns a token that you can specify in a subsequent call to get the next set of results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

### nextToken

The token for the next set of items to return. (You received this token from a previous call.)

Type: String

Pattern: `[A-Za-z0-9\-\_]*={0,3}`

Required: No

## Response Syntax

```
{
  "nextToken": "string",
  "toolList": [
```

```
{
  "description": "string",
  "id": "string",
  "latestVersionId": "string",
  "name": "string",
  "usedFor": "string"
}
]
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

The token to use when requesting the next set of items. If there are no additional items to return, the string is empty.

Type: String

Pattern: `[A-Za-z0-9\-\_]*={0,3}`

### toolList

Array of Tools objects

Type: Array of [ToolSummary](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerError**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### ThrottlingException

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `ListTools`.

### Example

#### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.ListTools
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "maxResults": 2
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
```

```
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "nextToken": "nextTokenToRetrieveNextPage",
  "toolList": [
    {
      "id": "PinpointDashboard",
      "latestVersionId": "1.0.0",
      "name": "Pinpoint Dashboard",
      "usedFor": "Listing customer's Pinpoint applications across regions.",
      "description": "Lists Pinpoint applications across regions.",
    },
    {
      "id": "CloudWatchLogGroupLookup",
      "latestVersionId": "1.0.1",
      "name": "CloudWatch LogGroup Lookup",
      "usedFor": "Investigating CloudWatch log groups using specified Log group
name",
      "description": "Investigate CloudWatch log groups.",
    }
  ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)





# StartExecution

Triggers new execution of a specific tool.

## Request Syntax

```
{
  "executionParameters": "string",
  "roleArn": "string",
  "storageRegion": "string",
  "tags": [
    {
      "key": "string",
      "value": "string"
    }
  ],
  "targetRegions": [ "string" ],
  "toolId": "string",
  "toolVersionId": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### [executionParameters](#)

Tool Execution Parameter.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32764.

Required: No

### [roleArn](#)

ARN of Service Role passed which will be used to execute Diagnostic Tool

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Pattern: `arn:aws[a-z0-9-]*:iam::\d{12}:role\[w\-\.\@+=,\]{1,65}`

Required: Yes

### storageRegion

Select AWS Region where you want to store execution output from available options

Type: String

Valid Values: `us-east-2 | us-west-2 | eu-west-1`

Required: Yes

### tags

Optional metadata that you assign to a resource. You can specify a maximum of five tags for an execution. Tags enable you to categorize a resource in different ways, such as by purpose, owner, or environment.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Required: No

### targetRegions

List of AWS Region Diagnostic Tool service allowed to query AWS Services during tool execution

Type: Array of strings

Required: Yes

### toolId

Unique-id of Diagnostic Tool which you want to execute

Type: String

Length Constraints: Minimum length of 8. Maximum length of 64.

Pattern: `\w+`

Required: Yes

### toolVersionId

Associate Version id for tool which you to execution

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5}].[0-9]{1,5}].[0-9]{1,5}`

Required: No

## Response Syntax

```
{
  "execution": {
    "creationTime": number,
    "executionId": "string",
    "parameters": "string",
    "requesterArn": "string",
    "requesterId": "string",
    "requestState": "string",
    "roleArn": "string",
    "status": "string",
    "storageRegion": "string",
    "targetRegions": [ "string" ],
    "toolId": "string",
    "toolVersionId": "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### execution

Execution Information Metadata

Type: [Execution](#) object

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **AccessDeniedException**

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### **InternalServerErrorException**

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### **ResourceNotFoundException**

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### **ServiceQuotaExceededException**

The request failed because it would cause a service quota to be exceeded.

HTTP Status Code: 400

### **ThrottlingException**

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### **ValidationException**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `StartExecution`.

## Example

### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.StartExecution
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "toolId": "EC2SystemsManager",
  "toolVersionId": "1.0.0",
  "roleArn": "<sample role>",
  "targetRegions": [
    "us-east-1"
  ],
  "storageRegion": "us-east-2",
  "parameters": {"apis": [true, true, true, false, false, false, false, false]}
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{
  "execution": {
    "creationTime": 1700601,
    "executionId": "e-aaaaaaaa",
    "requestState": "SUBMITTED",
    "requesterArn": "requesterArn",
    "requesterId": "user-id",
    "roleArn": "<sample role>",
  }
}
```

```
    "status": "CREATED",
    "storageRegion": "us-east-2",
    "tags": [],
    "targetRegions": [
      "us-east-1"
    ],
    "toolId": "EC2SystemsManager",
    "toolVersionId": "1.0.0"
  }
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# TagResource

Adds or overwrites one or more tags for the specified resource.

## Request Syntax

```
{
  "identifier": "string",
  "tags": [
    {
      "key": "string",
      "value": "string"
    }
  ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

The resource ID you want to tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

### tags

One or more tags. The value parameter is required.

Type: Array of [Tag](#) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Required: Yes



## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### InternalServerErrorException

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### ResourceNotFoundException

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### ThrottlingException

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### TooManyTagsException

Request exceeded maximum number of tag allowed..

HTTP Status Code: 400

### ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of TagResource.

## Example

### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.TagResource
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "identifier": "e-aaaaaaaaaa"
  "tags": [{
    "key": "department",
    "value": "support"
  }]
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# UntagResource

Removes tag keys from the specified resource.

## Request Syntax

```
{
  "identifier": "string",
  "tagKeys": [ "string" ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

### identifier

The resource identifier you want to remove tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

### tagKeys

Tag keys that you want to remove from the specified resource.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `(?!(?i)aws:).*`

Required: Yes

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 400

### InternalServerErrorException

The request failed because of an internal error. Try your request again later

HTTP Status Code: 500

### ResourceNotFoundException

The request failed because it references a resource that doesn't exist.

HTTP Status Code: 400

### ThrottlingException

The request failed because it exceeded a throttling quota.

HTTP Status Code: 400

### TooManyTagsException

Request exceeded maximum number of tag allowed..

HTTP Status Code: 400

### ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## Examples

This example illustrates one usage of `UntagResource`.

## Example

### Using AWS JSON protocol (default)

#### Sample Request

```
POST / HTTP/1.1
Host: ts.us-east-2.amazonaws.com
X-Amz-Target: Troubleshooting.UntagResource
Content-Type: application/x-amz-json-1.0
X-Amz-Date: <Date>
Authorization: <AuthParams>
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
{
  "identifier": "e-aaaaaaaaaa"
  "tagKeys": [
    "department"
  ]
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <requestId>
Content-Length: 0
Date: <Date>
Content-Type: application/x-amz-json-1.0
{}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# Data Types

The AWS Diagnostic Service API contains several data types that various actions use. This section describes each data type in detail.

## Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [Execution](#)
- [ExecutionSummary](#)
- [Tag](#)
- [Tool](#)
- [ToolSummary](#)
- [ToolVersion](#)
- [ValidationExceptionField](#)



# Execution

Information about Diagnostic Tool Execution.

## Contents

### **creationTime**

Execution Creation timestamp in Epoch format.

Type: Timestamp

Required: Yes

### **executionId**

Resource Identifier for Execution initiated via StartExecution Operation.

Type: String

Length Constraints: Fixed length of 12.

Pattern: e-[a-zA-Z0-9]{10}

Required: Yes

### **parameters**

Diagnostic Tool Execution Input parameters.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32764.

Required: Yes

### **requesterArn**

IAM User ARN of user/role who initiated the Execution.

Type: String

Required: Yes

### **requesterId**

IAM User ID who initiated the Execution.

Type: String

Required: Yes

### **requestState**

Indicate state of Execution.

Type: String

Valid Values: ACCEPTED | SUBMITTED | REJECTED

Required: Yes

### **roleArn**

The Execution role used by AWS Diagnostic Tools during Diagnostic Tool invocation.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Pattern: `arn:aws[a-z0-9-]*:iam::\d{12}:role\[/\w\-\./@+=,]{1,65}`

Required: Yes

### **storageRegion**

The AWS Regions where Execution result will be stored.

Type: String

Valid Values: us-east-2 | us-west-2 | eu-west-1

Required: Yes

### **toolId**

Tool Id associated with Execution.

Type: String

Length Constraints: Minimum length of 8. Maximum length of 64.

Pattern: `\w+`

Required: Yes

## **toolVersionId**

Diagnostic Tool version identifier associated with tool artifact.

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5}].[0-9]{1,5}].[0-9]{1,5}`

Required: Yes

## **status**

The Execution Status.

Type: String

Valid Values: `CREATED` | `IN_PROGRESS` | `SUCCESSFUL` | `FAILED` | `CANCELLED` | `TIMED_OUT`

Required: No

## **targetRegions**

The AWS Regions targeted by the Diagnostic Tool Execution.

Type: Array of strings

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# ExecutionSummary

Information about the Execution.

## Contents

### creationTime

Describe Execution Creation Time.

Type: Timestamp

Required: Yes

### executionId

Resource Identifier for Execution initiated via StartExecution Operation.

Type: String

Length Constraints: Fixed length of 12.

Pattern: e-[a-zA-Z0-9]{10}

Required: Yes

### requesterArn

The Execution role used by AWS Diagnostic Tools during Diagnostic Tool invocation.

Type: String

Required: Yes

### requesterId

IAM User ID who initiated the Execution.

Type: String

Required: Yes

### requestState

Indicate state of Execution.

Type: String

Valid Values: ACCEPTED | SUBMITTED | REJECTED

Required: Yes

### **roleArn**

The Execution role used by AWS Diagnostic Tools during Diagnostic Tool invocation.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Pattern: `arn:aws[a-z0-9-]*:iam::\d{12}:role\[/\w\-\./@+=,]{1,65}`

Required: Yes

### **storageRegion**

The AWS Regions targeted by the Diagnostic Tool Execution.

Type: String

Valid Values: us-east-2 | us-west-2 | eu-west-1

Required: Yes

### **toolId**

Tool Id associated with Execution.

Type: String

Length Constraints: Minimum length of 8. Maximum length of 64.

Pattern: `\w+`

Required: Yes

### **toolVersionId**

Diagnostic Tool version identifier associated with tool artifact.

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5}].[0-9]{1,5}].[0-9]{1,5}`

Required: Yes

### **status**

The Execution Status.

Type: String

Valid Values: `CREATED` | `IN_PROGRESS` | `SUCCESSFUL` | `FAILED` | `CANCELLED` | `TIMED_OUT`

Required: No

### **targetRegions**

Describe Execution Target Regions.

Type: Array of strings

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Tag

Tag is key and value pair that act as metadata for organizing your AWS resources.

## Contents

### key

Tag Key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: (?!(?i)aws:).\*

Required: Yes

### value

Value for the Tag key.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Tool

Information about a Diagnostic Tool.

## Contents

### id

Unique Identifier for Diagnostic Tool .

Type: String

Length Constraints: Minimum length of 8. Maximum length of 64.

Pattern: `\w+`

Required: Yes

### latestVersionId

Field specifying the latest available version of the artifact associated with the Diagnostic Tool.

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5}].[0-9]{1,5}].[0-9]{1,5}`

Required: Yes

### name

Describes the Diagnostic Tool Name.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 256.

Pattern: `[\ \w]+`

Required: Yes

### versions

Array of versions object associated with the Tool.



Type: Array of [ToolVersion](#) objects

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# ToolSummary

Information about the Diagnostic Tool.

## Contents

### description

Description the Diagnostic Tool.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 512.

Required: Yes

### id

Unique Identifier for Diagnostic Tool.

Type: String

Length Constraints: Minimum length of 8. Maximum length of 64.

Pattern: `\w+`

Required: Yes

### latestVersionId

Version Identifier associated to the latest version of Diagnostic Tool

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5}.[0-9]{1,5}.[0-9]{1,5}`

Required: Yes

### name

Describes the Diagnostic Tool Name.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 256.

Pattern: `[\ \w]+`

Required: Yes

### **usedFor**

List of use cases where these Diagnostic Tool can be used.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 512.

Required: Yes

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# ToolVersion

Information about the specific Diagnostic Tool Version.

## Contents

### description

Version description.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 512.

Required: Yes

### parametersSchema

Diagnostic Tool input parameter schema.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32764.

Required: Yes

### usedFor

List of use cases where these Diagnostic Tool can be used.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 512.

Required: Yes

### versionId

Version Identifier associated to specific version of Diagnostic Tool artifact.

Type: String

Length Constraints: Minimum length of 5. Maximum length of 17.

Pattern: `[0-9]{1,5} . [0-9]{1,5} . [0-9]{1,5}`

Required: Yes

## labels

Additional Diagnostic Tool metadata.

Type: Array of strings

Valid Values: GLOBAL\_TOOL

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# ValidationExceptionField

Returns information about a field passed inside a request that resulted in an exception.

## Contents

### message

Message describing why the field failed validation.

Type: String

Required: Yes

### name

The name of the field which failed the validation.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see [Signing AWS API requests](#) in the *IAM User Guide*.

## Action

The action to be performed.

Type: string

Required: Yes

## Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

## X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

## X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4\_request"). The value is expressed in the following format: *access\_key/YYYYMMDD/region/service/aws4\_request*.

For more information, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

### **X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Elements of an AWS API request signature](#) in the *IAM User Guide*.

Type: string

Required: Conditional

### **X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see [AWS services that work with IAM](#) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

### **X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.



Type: string

Required: Conditional

### **X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

# Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

## **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

## **IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

## **InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

## **InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

## **InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

## **NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

## **OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

**RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

**ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

**ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

**ValidationError**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400