

Create a service hook subscription programmatically

Article • 02/07/2025

Azure DevOps Services | Azure DevOps Server 2022 - Azure DevOps Server 2019

Using the [Subscriptions REST APIs](#), you can programmatically create a subscription that performs an action on an external/consumer service when a specific event occurs in an Azure DevOps project. For example, you can create a subscription to notify your service when a build fails.

Supported events:

- Build completed
- Code pushed (for Git projects)
- Pull request create or updated (for Git projects)
- Code checked in (TFVC projects)
- Work item created, updated, deleted, restored or commented on

You can configure filters on your subscriptions to control which events trigger an action. For example, you can filter the build completed event based on the build status. For a complete set of supported events and filter options, see the [Event reference](#).

For a complete set of supported consumer services and actions, see the [Consumer reference](#).

Prerequisites

 Expand table

Category	Requirements
Project access	Project member .
Data	<ul style="list-style-type: none">- Project ID. Use the Project REST API to get the project ID.- Event ID and settings. See the Event reference.- Consumer and action IDs and settings. See the Consumer reference.

Create the request

Construct the body of the HTTP POST request to create the subscription based on the project ID, event, consumer, and action.

See the following example request for creating a subscription that causes a build event to POST to `https://myservice/event` when the build `WebSite.CI` fails.

Request

JavaScript

```
{
  "publisherId": "tfs",
  "eventType": "build.complete",
  "resourceVersion": "1.0",
  "consumerId": "webHooks",
  "consumerActionId": "httpRequest",
  "publisherInputs": {
    "buildStatus": "failed",
    "definitionName": "WebSite.CI",
    "projectId": "56479caf-1eeb-4bca-86ab-aaa6f29399d9",
  },
  "consumerInputs": {
    "url": " https://myservice/event"
  },
}
```

We highly recommend using secure HTTPS URLs for the security of the private data in the JSON object.

Response See the following response to the request to create the subscription:

JavaScript

```
{
  "id": "aaaa0a0a-bb1b-cc2c-dd3d-eeeeee4e4e4e",
  "url":
  "https://dev.azure.com/fabrikam/DefaultCollection/_apis/hooks/subscriptions/aaaa0a0a-bb1b-cc2c-dd3d-eeeeee4e4e4e",
  "publisherId": "tfs",
  "eventType": "build.complete",
  "resourceVersion": "1.0",
  "consumerId": "webHooks",
  "consumerActionId": "httpRequest",
  "createdBy": {
    "id": "00ca946b-2fe9-4f2a-ae2f-40d5c48001bc"
  },
}
```

```
{
  "createdDate": "2014-03-28T16:10:06.523Z",
  "modifiedBy": {
    "id": "1c4978ae-7cc9-4efa-8649-5547304a8438"
  },
  "modifiedDate": "2014-04-25T18:15:26.053Z",
  "publisherInputs": {
    "buildStatus": "failed",
    "definitionName": "WebSite.CI",
    "hostId": "17f27955-99bb-4861-9550-f2c669d64fc9",
    "projectId": "56479caf-1eeb-4bca-86ab-aaa6f29399d9",
    "tfsSubscriptionId": "29cde8b4-f37e-4ef9-a6d4-d57d526d82cc"
  },
  "consumerInputs": {
    "url": "http://myservice/event"
  }
}
```

If the subscription request fails, you get an HTTP response code of 400 with a message that has further details.

What happens when the event occurs?

When an event occurs, all enabled subscriptions in the project are evaluated, and the consumer action is performed for all matching subscriptions.

Resource versions (advanced)

Resource versioning is applicable when an API is in preview. For most scenarios, specifying 1.0 as the resource version is the safest route.

The event payload sent to certain consumers, like Webhooks, Azure Service Bus, and Azure Storage, includes a JSON representation of subject resource (for example, a build or work item). The representation of this resource can have different forms or versions.

You can specify the version of the resource that you want to have sent to the consumer service via the `resourceVersion` field on the subscription. The resource version is the same as the [API version](#). Not specifying a resource version means "latest released". You should always specify a resource version, which ensures a consistent event payload over time.

FAQs

Q: Are there services that I can subscribe to manually?

A: Yes. For more information about the services that you can subscribe to from the administration page for a project, see the [Overview](#).

Q: Are there C# libraries that I can use to create subscriptions?

A: No, but here's a sample to help you get started.

cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Mvc;

namespace Microsoft.Samples.VisualStudioOnline
{
    public class ServiceHookEventController : Controller
    {
        // POST: /ServiceHookEvent/workitemcreated
        [HttpPost]
        public HttpResponseMessage WorkItemCreated(Content workItemEvent)
        {
            //Grabbing the title for the new workitem
            var value = RetrieveFieldValue("System.field",
workItemEvent.Resource.Fields);

            //Acknowledge event receipt
            return new HttpResponseMessage(HttpStatusCode.OK);
        }

        /// <summary>
        /// Gets the value for a specified work item field.
        /// </summary>
        /// <param name="key">Key used to retrieve matching value</param>
        /// <param name="fields">List of fields for a work item</param>
        /// <returns></returns>
        public String RetrieveFieldValue(String key, IList<FieldInfo> fields)
        {
            if (String.IsNullOrEmpty(key))
                return String.Empty;

            var result = fields.Single(s => s.Field.RefName == key);
```

```

        return result.Value;
    }
}

public class Content
{
    public String SubscriptionId { get; set; }

    public int NotificationId { get; set; }

    public String EventType { get; set; }

    public WorkItemResource Resource { get; set; }
}

public class WorkItemResource
{
    public String UpdatesUrl { get; set; }

    public IList<FieldInfo> Fields { get; set; }

    public int Id { get; set; }

    public int Rev { get; set; }

    public String Url { get; set; }

    public String WebUrl { get; set; }
}

public class FieldInfo
{
    public FieldDetailedInfo Field { get; set; }

    public String Value { get; set; }
}

public class FieldDetailedInfo
{
    public int Id { get; set; }

    public String Name { get; set; }

    public String RefName { get; set; }
}
}

```

Related articles

- [Authorize service hooks](#)
- [Consumers](#)
- [Events](#)
- [Troubleshooting and FAQs](#)