# Ninja RMM Public API v2.0.5 Webhooks

Webhooks provide instant notifications about activities and other events via HTTP requests to your web services.

Steps to configure webhook integration:

1. Create and publish a webservice that will accept POST requests from NinjaRMM API Webhooks
2. Configure NinjaRMM Webhook via API

## 1. Creating webservice for NinjaRMM API Webhook

NinjaRMM webhook activity notifications are delivered via POST requests to your webservice. Schema is very similar to **Activity** object for API v2.0.

```
{
  "id": 12938,
  "activityTime": 1592600612,
  "deviceId": 100,
  "severity": "NONE | MINOR | MODERATE | MAJOR | CRITICAL",
  "priority": "NONE | LOW | MEDIUM | HIGH",
  "seriesUid": "Activity series UID (job/condition UID)",
  "activityType": "ACTIONSET",
  "statusCode": "START_REQUESTED",
  "status": "string",
  "activityResult": "SUCCESS | FAILURE | UNSUPPORTED | UNCOMPLETED",
  "sourceConfigUid": "Source configuration/policy element reference",
  "sourceName": "Source configuration/policy element name",
  "subject": "Activity subject",
  "userId": 0,
  "message": "Activity message in selected language/locale",
  "type": "Activity type description",
  "data": { … },
  "device": { … }
```

Exact specification can be found in *webhook.yaml* file. This file can also be used to generate a stub for webservice in a programming language of your choice:

- Online using [https://editor.swagger.io/](https://editor.swagger.io/)
- CLI with OpenAPI Generator: `openapi-generator generate -I webhook.yaml -g kotlin-server -o ninjarmm-webhook`

Generated server code will have an empty endpoint returning **501 Not Implemented**.
You will need to change the URL and provide implementation for your Webhook.

## 2. Configuring NinjaRMM webhook URL

Once you have deployed your webservice you need to configure NinjaRMM account to push activity notifications by making a PUT request to **/v2/webhook** with the following content:

```
{
    "url": "https://callbacks.mymsphost.com/webhooks/ninjarmm/activity",
    "expand": ["device","organization","location"],
    "activities": {
        "CONDITION": [
            "*"
        ],
        "SYSTEM": [
            "*"
        ]
    }
}
```

Where:

**url** – URL where your webservice endpoint accepting activity notifications

**expand** – which objects to expand. By default NinjaRMM API will only supply **deviceId**. If your application does not maintain a cache of core entities (devices, organizations, locations) you can configure webhook to expand those references in notifications. This will result in notifications' payloads to become larger. We encourage you to maintain local cache of core entities that can be loaded via Public API v2.0.

**activities** – activity filter object. Each element in object should be named after type of activity (*see webhook.yaml*) and value array `["*"]` (This is reserved for future more granular activity filtering).

Note: you can use https://app.ninjarmm.com/apidocs/ to perform this **PUT /v2/webhook** request