

Testen des NPM-Registers mit Roh-HTTP-Anfragen

Um Ihr NPM-Registry mit Roh-HTTP-Anfragen mit Tools wie `curl` oder `Insomnia` zu testen, können Sie dessen Funktionalität überprüfen, indem Sie die Konnektivität, Authentifizierung und die Fähigkeit zur Bereitstellung von Paketdaten überprüfen –alles ohne auf den NPM-Client zurückzugreifen. Da Ihr Registry einen Benutzernamen und ein Passwort erfordert, müssen Sie die Authentifizierung manuell in Ihren Anfragen verwalten. Hier ist eine vollständige Anleitung zum Testen Ihres Registries mit Roh-HTTP-Anfragen, vorausgesetzt, es handelt sich um ein NPM-kompatibles Registry (z. B. Verdaccio, Nexus oder eine benutzerdefinierte Implementierung), das Token-basierte oder Basis-Authentifizierung unterstützt.

Schritte zum Testen des Registries mit Roh-HTTP-Anfragen

1. Authentifizieren und Token erhalten (falls erforderlich)

Die meisten NPM-Registries verwenden eine Token-basierte Authentifizierung, bei der Sie sich anmelden müssen, um ein Token zu erhalten, bevor Sie weitere Anfragen senden. Einige Registries unterstützen möglicherweise auch eine direkte Basis-Authentifizierung. Hier ist, wie Sie sich mit `curl` authentifizieren:

Mit `curl` Anmelden Senden Sie eine PUT-Anfrage an das Authentifizierungs-Endpoint des Registries, um ein Token zu erhalten:

```
curl -X PUT \
  -H "Content-Type: application/json" \
  -d '{"name": "<username>", "password": "<password>"}' \
  <registry-url>/-/user/org.couchdb.user:<username>
```

- **Ersetzen Sie:**

- `<username>`: Ihr Registry-Benutzername.
- `<password>`: Ihr Registry-Passwort.
- `<registry-url>`: Die vollständige URL Ihres Registries (z. B. `https://my-registry.example.com`).

- **Erwartete Antwort:** Bei Erfolg erhalten Sie eine JSON-Antwort mit einem Token:

```
{
  "token": "your-auth-token"
}
```

- **Token speichern:** Kopieren Sie den Wert `your-auth-token` für die Verwendung in nachfolgenden Anfragen.

Hinweis: Wenn Ihr Registry ein anderes Authentifizierungs-Endpoint oder -Verfahren verwendet (z. B. Basis-Auth oder eine benutzerdefinierte API), überprüfen Sie dessen Dokumentation. Wenn es Basis-Auth direkt unterstützt, können Sie diesen Schritt überspringen und `-u "<username>:<password>"` in späteren Anfragen hinzufügen.

2. Ping des Registries

Testen Sie die grundlegende Konnektivität zum Registry, indem Sie eine GET-Anfrage an seine Root-URL oder ein Ping-Endpoint senden.

Mit `curl` Ping

```
curl -H "Authorization: Bearer your-auth-token" <registry-url>
```

- **Ersetzen Sie:**

- `your-auth-token`: Das Token aus Schritt 1.
- `<registry-url>`: Ihre Registry-URL.

- **Erwartete Antwort:** Eine erfolgreiche Antwort (HTTP 200) könnte die Startseite des Registries oder eine einfache Statusmeldung (z. B. `{"db_name": "registry"}` für CouchDB-basierte Registries) zurückgeben.

- **Alternative:** Einige Registries bieten ein `/-/ping`-Endpoint:

```
curl -H "Authorization: Bearer your-auth-token" <registry-url>/-/ping
```

Wenn Basis-Auth verwendet wird: Wenn Ihr Registry keine Tokens verwendet und Basis-Authentifizierung unterstützt:

```
curl -u "<username>:<password>" <registry-url>
```

3. Paketmetadaten abrufen

Überprüfen Sie, ob das Registry Paketmetadaten bereitstellen kann, indem Sie Details für ein bestimmtes Paket anfordern.

Mit curl Metadaten abrufen

```
curl -H "Authorization: Bearer your-auth-token" <registry-url>/<package-name>
```

- **Ersetzen Sie:**

- <package-name>: Ein Paket, von dem Sie wissen, dass es in Ihrem Registry vorhanden ist (z. B. lodash, wenn es das öffentliche Registry proxyt, oder ein privates Paket wie my-org-utils).

- **Erwartete Antwort:** Ein JSON-Objekt mit den Metadaten des Pakets, einschließlich Versionen, Abhängigkeiten und Tarball-URLs. Zum Beispiel:

```
{
  "name": "lodash",
  "versions": {
    "4.17.21": {
      "dist": {
        "tarball": "<registry-url>/lodash/-/lodash-4.17.21.tgz"
      }
    }
  }
}
```

Wenn Basis-Auth verwendet wird:

```
curl -u "<username>:<password>" <registry-url>/<package-name>
```

- **Erfolg:** Eine 200 OK-Antwort mit Metadaten bestätigt, dass das Registry Paketdaten korrekt bereitstellt.
-

4. Pakettarball herunterladen (optional)

Um das Registry vollständig zu testen, laden Sie einen Pakettarball herunter, um sicherzustellen, dass es die tatsächlichen Paketdateien liefern kann.

Mit curl Tarball herunterladen

1. Aus den Metadaten in Schritt 3 die tarball-URL für eine bestimmte Version finden (z. B. <registry-url>/lodash/-/lodash-4.17.21.tgz).
2. Laden Sie es herunter:

```
curl -H "Authorization: Bearer your-auth-token" -O <tarball-url>
```

- **Ersetzen Sie:** <tarball-url> durch die URL aus den Metadaten.
- **-O-Flag:** Speichert die Datei mit ihrem ursprünglichen Namen (z. B. `lodash-4.17.21.tgz`).
- **Wenn Basis-Auth verwendet wird:**

```
curl -u "<username>:<password>" -O <tarball-url>
```

- **Erfolg:** Die Datei wird erfolgreich heruntergeladen und Sie können sie extrahieren (z. B. mit `tar -xzf <filename>`), um deren Inhalte zu überprüfen.

Testen mit Insomnia

Wenn Sie ein GUI-Tool wie Insomnia bevorzugen, befolgen Sie diese Schritte:

1. Authentifizierung einrichten

- Erstellen Sie eine neue Anfrage in Insomnia.
- Gehen Sie zur **Auth**-Registerkarte:
 - **Bearer Token:** Wenn Sie ein Token in Schritt 1 erhalten haben, wählen Sie "Bearer Token" und fügen Sie `your-auth-token` ein.
 - **Basic Auth:** Wenn das Registry Basis-Auth verwendet, wählen Sie "Basic Auth" und geben Sie Ihren `<username>` und `<password>` ein.

2. Ping des Registries

- **Methode:** GET
- **URL:** <registry-url> oder <registry-url>/-/ping
- Klicken Sie auf **Senden**.
- **Erwartete Antwort:** Ein 200 OK-Status mit einer einfachen Antwortkörper.

3. Paketmetadaten abrufen

- **Methode:** GET
- **URL:** <registry-url>/<package-name>
- Stellen Sie sicher, dass die Authentifizierung in der Auth-Registerkarte eingestellt ist.
- Klicken Sie auf **Senden**.
- **Erwartete Antwort:** Ein 200 OK-Status mit Paketmetadaten in JSON.

4. Tarball herunterladen

- **Methode:** GET
 - **URL:** Die Tarball-URL aus den Metadaten (z. B. `<registry-url>/<package-name>/-/<package-name>-<version>.tgz`)
 - Klicken Sie auf **Senden und Herunterladen**, um die Datei lokal zu speichern.
 - **Erfolg:** Die Datei wird heruntergeladen, was bestätigt, dass das Registry Pakete bereitstellt.
-

Fehlerbehebung

- **401 Unauthorized:**
 - Überprüfen Sie Ihr Token oder Ihre Anmeldeinformationen.
 - Stellen Sie sicher, dass der `Authorization`-Header oder die Basis-Auth korrekt formatiert ist.
 - **404 Not Found:**
 - Überprüfen Sie, ob das Paket im Registry vorhanden ist.
 - Wenn Sie öffentliche Pakete testen (z. B. `lodash`), stellen Sie sicher, dass das Registry das öffentliche NPM-Registry proxyt.
 - **Verbindungsfehler:**
 - Überprüfen Sie die `<registry-url>` (z. B. fügen Sie `https://` hinzu, falls erforderlich).
 - Bei HTTPS-Registries, wenn Sie SSL-Fehler mit `curl` erhalten, fügen Sie `-k` hinzu, um Zertifikatsprüfungen zu umgehen (nur für Tests).
 - **Benutzerdefiniertes Registry-Verhalten:**
 - Wenn diese Schritte fehlschlagen, konsultieren Sie die Dokumentation Ihres Registries für spezifische Endpoints oder Authentifizierungsmethoden.
-

Schlussfolgerung

Durch die Befolgung dieser Schritte mit `curl` oder `Insomnia` können Sie die Konnektivität, Authentifizierung und Paketbereitstellungsfähigkeiten Ihres NPM-Registries mit Roh-HTTP-Anfragen testen. Beginnen Sie mit der Authentifizierung (falls erforderlich), dann pingen Sie das Registry, rufen Sie Metadaten für ein bekanntes Paket ab und laden Sie optional einen Tarball herunter. Dieser Ansatz stellt sicher, dass Ihr Registry auf der HTTP-Ebene vollständig funktionsfähig ist.