

## npm कनेक्शन पर परीक्षण

आपके npm रजिस्ट्री का काम करने के लिए, विशेष रूप से जब यह एक यूजरनेम और पासवर्ड की आवश्यकता होती है, आप कुछ सरल कदमों का पालन कर सकते हैं ताकि इसकी कनेक्टिविटी, ऑथेंटिकेशन और पैकेजों को सर्व करने की क्षमता की जांच कर सकें। क्योंकि यह एक प्राइवेट रजिस्ट्री है (क्रेडेंशियल्स से संकेतित), आप ऑथेंटिकेशन को सही ढंग से सेट अप करने की सुनिश्चितता करनी चाहिए पहले पैकेज इंस्टॉलेशन की जांच करने से।

### रजिस्ट्री को टेस्ट करने के कदम

#### 1. ऑथेंटिकेशन सेट अप करें

- क्योंकि रजिस्ट्री एक यूजरनेम और पासवर्ड की आवश्यकता होती है, आप अपने npm क्लाइंट को ऑथेंटिकेट करने के लिए लॉग इन कर सकते हैं।

- अपने टर्मिनल में निम्नलिखित कमांड चलाएं, <registry-url> को अपने रजिस्ट्री के वास्तविक npm से बदलें:

```
npm login --registry=<registry-url>
```

- आपको अपना यूजरनेम और पासवर्ड (और संभवतः एक ईमेल, रजिस्ट्री के आधार पर) दर्ज करने के लिए प्रॉम्प्ट किया जाएगा। एक बार पूरा होने पर, यह एक ऑथेंटिकेशन टोकन को आपके .npmrc फ़ाइल में स्टोर करता है, जिससे रजिस्ट्री तक पहुंच मिलती है।

#### 2. रजिस्ट्री कनेक्शन की जांच करें

- npm ping कमांड का उपयोग करें ताकि रजिस्ट्री पहुंचने योग्य है:

```
npm ping --registry=<registry-url>
```

- अगर सफल, यह कमांड यह सत्यापित करता है कि रजिस्ट्री सर्वर ऊपर है और जवाब दे रहा है। आप आम तौर पर "npm ping: 200 OK" जैसा आउटपुट देखेंगे। अगर यह विफल हो जाता है, तो संभवतः कनेक्शन समस्या हो सकती है या npm गलत हो सकता है।

#### 3. ऑथेंटिकेशन की जांच करें

- सुनिश्चित करने के लिए कि आपका यूजरनेम और पासवर्ड सही ढंग से सेट अप हैं, npm whoami कमांड का उपयोग करें:

```
npm whoami --registry=<registry-url>
```

- अगर ऑथेंटिकेशन सफल है, तो यह आपका यूजरनेम वापस कर देगा। अगर यह विफल हो जाता है या एक त्रुटि (जैसे, "npm whoami: 401 Unauthorized") वापस करता है, तो अपने क्रेडेंशियल्स या लॉगिन कदम को दोबारा जांचें।

#### 4. पैकेज इंस्टॉलेशन का टेस्ट करें

- एक पैकेज इंस्टॉल करने की कोशिश करें ताकि रजिस्ट्री पैकेजों को सर्व कर सके। क्योंकि यह एक प्राइवेट रजिस्ट्री है, आपको एक पैकेज इंस्टॉल करना होगा जिसे आप जानते हैं कि यह उस पर मौजूद है। हालांकि, अगर रजिस्ट्री सार्वजनिक npm रजिस्ट्री को प्रॉक्सी करता है (प्राइवेट रजिस्ट्री जैसे npmjs.org के लिए एक आम सेटअप), आप लोकप्रिय सार्वजनिक पैकेजों के साथ टेस्ट कर सकते हैं।

- उदाहरण कमांड:

```
npm install <package-name> --registry=<registry-url>
```

- <package-name> को अपने रजिस्ट्री पर उपलब्ध पैकेज के साथ बदलें (नीचे पैकेज सुझावों के बारे में अधिक जानें)।

## कुछ npm पैकेजों को ट्राई करें

क्योंकि यह एक प्राइवेट रजिस्ट्री है, मैं ठीक से जान सकता हूँ कि कौन से पैकेज उपलब्ध हैं। हालांकि, आम सीनारियो के आधार पर कुछ सुझाव हैं:

### □ अगर रजिस्ट्री सार्वजनिक npm रजिस्ट्री को प्रॉक्सी करता है:

- कई प्राइवेट रजिस्ट्री सार्वजनिक रजिस्ट्री को मिरर करने के लिए कॉन्फिगर किए जाते हैं, जिससे ऑथेंटिकेशन के बाद सार्वजनिक पैकेजों तक पहुंच मिलती है। इस मामले में, आप लोकप्रिय सार्वजनिक पैकेजों को इंस्टॉल करने की कोशिश कर सकते हैं:

★ **lodash**: एक लोकप्रिय यूटिलिटी लाइब्रेरी।

```
npm install lodash --registry=<registry-url>
```

★ **express**: npm के लिए एक व्यापक रूप से उपयोग किया जाने वाला वेब फ्रेमवर्क।

```
npm install express --registry=<registry-url>
```

★ **react**: उपयोगकर्ता इंटरफ़ेस बनाने के लिए एक लोकप्रिय लाइब्रेरी।

```
npm install react --registry=<registry-url>
```

- अगर ये सफलतापूर्वक इंस्टॉल हो जाते हैं, तो यह सत्यापित करता है कि रजिस्ट्री काम कर रहा है और पैकेजों को सर्व कर सकता है।

### □ अगर रजिस्ट्री केवल प्राइवेट पैकेजों को होस्ट करता है:

- आपको एक पैकेज इंस्टॉल करना होगा जिसे आप जानते हैं कि यह आपके प्राइवेट रजिस्ट्री पर मौजूद है। अपने टीम या रजिस्ट्री के दस्तावेज/वेब इंटरफ़ेस (अगर उपलब्ध है) से एक पैकेज नाम की जांच करें। उदाहरण के लिए:

★ अगर एक पैकेज `my-org-utils` है, तो कोशिश करें:

```
npm install my-org-utils --registry=<registry-url>
```

- बिना विशिष्ट पैकेज नामों के, मैं ठीक से प्राइवेट पैकेजों की सूची नहीं दे सकता, इसलिए अपने संगठन के लिए एक रिलेवेंट पैकेज का उपयोग करें।

---

## नोट्स और ट्रबलशूटिंग

- **सफलता संकेतक**: अगर `npm ping` काम करता है, `npm whoami` आपका यूजरनेम वापस करता है, और एक पैकेज बिना त्रुटि के इंस्टॉल होता है, तो आपका रजिस्ट्री सही ढंग से काम कर रहा है।

### □ आम समस्याएं:

- **ऑथेंटिकेशन त्रुटियां**: अगर `npm whoami` विफल हो जाता है, तो `npm login` को फिर से चलाएं।
- **पैकेज नहीं मिला**: सुनिश्चित करें कि पैकेज रजिस्ट्री पर मौजूद है। अगर सार्वजनिक पैकेज का उपयोग किया जा रहा है और यह विफल हो जाता है, तो आपका रजिस्ट्री संभवतः सार्वजनिक रजिस्ट्री को प्रॉक्सी नहीं करता है।
- **कनेक्शन समस्याएं**: अगर `npm ping` विफल हो जाता है, तो `<registry-url>` और अपने नेटवर्क सेटिंग्स को दोबारा जांचें।

- **ग्लोबल बदलाव से बचें**: अपने ग्लोबल npm कॉन्फिगरेशन को प्रभावित करने से बचने के लिए, आप एक नए डायरेक्टरी में एक स्थानीय `.npmrc` फ़ाइल के साथ टेस्ट कर सकते हैं, लेकिन ऊपर दिए गए कमांड `--registry` फ्लैग के साथ पर्याप्त हैं।

इन कदमों का पालन करके और सुझाए गए पैकेजों को ट्राई करके, आप यह सत्यापित कर सकते हैं कि आपका रजिस्ट्री ऑपरेशनल है और ऑथेंटिकेशन और पैकेज रिट्रीवल को सही ढंग से संभालता है। अगर आपके पास रजिस्ट्री से विशिष्ट पैकेज नाम हैं, तो इन सार्वजनिक उदाहरणों के बजाय उनका उपयोग करें।