# Accessing a Kerberized Cluster Through Envoy (Reverse Proxy)

## Objective

To enable secure access to a **Kerberos-protected cluster** (e.g., Hadoop NameNode UI or API) through **Envoy as a reverse proxy**, specifically for health check and service routing purposes.

## Overview

There are two approaches to enable Envoy to access and verify the availability of Kerberized backend services.

## Option 1: Whitelist Health Check URL (Bypass Kerberos)

### Description

This method involves configuring the Kerberos-protected backend to whitelist specific health check endpoints (e.g., /jmx, /healthz) so that they bypass Kerberos authentication. Envoy can then use its native health check functionality to monitor these endpoints without needing to include a Kerberos token.

### Note

**This option is not supported in our current Hadoop environment**, as it does not allow bypassing Kerberos for specific endpoints.

## Option 2: Send Kerberos Token via Custom Envoy Integration

### Description

In this method, Envoy sends an Authorization: Negotiate <Kerberos-token> header along with its health check or proxy request. This approach requires:

1. Integration of **custom code** in Envoy, such as:
   1. A **Lua filter**, or
   2. An **external authorization service (ext\_authz)**
2. The Kerberos token must be:
   1. Fetched using kinit and SPNEGO negotiation
   2. Stored securely
   3. Rotated regularly to avoid expiration and replay issues

### Considerations

1. Requires robust **token management and injection** mechanism
2. Involves **custom scripting** and logic for token refresh and validation
3. Requires **collaboration and support from the Envoy team** to implement and maintain the integration