Kea DHCP Manager

A modern, minimalist, and user-friendly web application for managing a Kea DHCP server on Debian-based systems. Built with Ruby and Sinatra, featuring a responsive dashboard-style interface for administrators.

Features

Core Functionality

- **Subnet Management**: Add and delete DHCP subnets with validation
- **IP Reservations**: Reserve specific IP addresses using MySQL database backend
- **Server Monitoring**: Real-time Kea server status and active lease monitoring
- **Configuration Management**: Validate, backup, and reload Kea configurations
- **Activity Logging**: Comprehensive logging of all DHCP management actions

User Interface

- **Responsive Design**: Modern, minimalist interface that works on all devices
- **Dashboard**: Overview of server status, active leases, and system health
- **Real-time Updates**: Auto-refreshing status indicators and lease information
- **Intuitive Forms**: User-friendly forms with validation and helpful feedback

Security & Reliability

- **Authentication**: Secure login system with session management
- **Input Validation**: Comprehensive validation for IP addresses, MAC addresses, and network configurations
- **Configuration Backup**: Automatic backup system with restore capabilities

- **Error Handling**: Robust error handling with detailed logging

Requirements

System Requirements

- **Operating System**: Debian-based Linux distribution (Ubuntu, Debian)
- **Ruby**: Version 3.0 or higher
- **MySQL**: Version 8.0 or higher
- **Kea DHCP**: Version 2.0 or higher

Dependencies

- Sinatra web framework
- MySQL2 database connector
- Sequel ORM
- Bootstrap 5 (included via CDN)

Installation

1. Install System Dependencies

```
# Update package list
sudo apt update

# Install Ruby and development tools
sudo apt install ruby ruby-dev build-essential

# Install MySQL server
sudo apt install mysql-server

# Install Kea DHCP server
sudo apt install kea-dhcp4-server kea-ctrl-agent

# Install Bundler gem
sudo gem install bundler
```

2. Download and Setup Application

```
# Clone or download the application
cd /opt
sudo git clone <repository-url> kea-dhcp-manager
cd kea-dhcp-manager

# Install Ruby dependencies
sudo bundle install

# Setup directories
rake install
```

3. Configure MySQL Database

```
# Secure MySQL installation
sudo mysql_secure_installation

# Setup database
rake setup_db
```

4. Configure Environment

```
# Copy environment configuration
cp .env.example .env

# Edit configuration file
nano .env
```

Update the `.env` file with your settings:

```
# Database Configuration
DB_HOST=localhost
DB_PORT=3306
DB_NAME=kea_dhcp
DB_USER=kea_user
DB_PASSWORD=your_secure_password

# Kea Configuration
KEA_CONFIG_PATH=/etc/kea/kea-dhcp4.conf
KEA_CONTROL_SOCKET=/tmp/kea4-ctrl-socket
KEA_LEASE_FILE=/var/lib/kea/dhcp4.leases

# Application Configuration
RACK_ENV=production
PORT=4567
LOG_LEVEL=INFO

# Security
SESSION_SECRET=your_session_secret_here
ADMIN_USERNAME=admin
ADMIN_PASSWORD=your_admin_password
```

5. Configure Kea DHCP

Ensure your Kea configuration includes a control socket:

```
{
    "Dhcp4": {
        "control-socket": {
            "socket-type": "unix",
            "socket-name": "/tmp/kea4-ctrl-socket"
        }
    }
}
```

Deployment

Development Mode

```
# Start in development mode
rake dev
```

Access the application at: `http://localhost:4567`

Production Deployment

Option 1: Direct Ruby Execution

```
# Start the application
rake start
```

Option 2: Using Systemd Service

Create a systemd service file:

```
sudo nano /etc/systemd/system/kea-dhcp-manager.service
```

```
[Unit]
Description=Kea DHCP Manager
After=network.target mysql.service kea-dhcp4-server.service

[Service]
Type=simple
User=www-data
Group=www-data
WorkingDirectory=/opt/kea-dhcp-manager
Environment=RACK_ENV=production
ExecStart=/usr/local/bin/bundle exec rackup -p 4567 -o 0.0.0.0
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
```

Enable and start the service:

```
sudo systemctl enable kea-dhcp-manager
sudo systemctl start kea-dhcp-manager
sudo systemctl status kea-dhcp-manager
```

Option 3: Using Nginx Reverse Proxy

Install and configure Nginx:

```
sudo apt install nginx

sudo nano /etc/nginx/sites-available/kea-dhcp-manager
```

```
server {
    listen 80;
    server_name your-server-domain.com;

    location / {
        proxy_pass http://127.0.0.1:4567;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }
}
```

Enable the site:

```
sudo ln -s /etc/nginx/sites-available/kea-dhcp-manager /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl restart nginx
```

Usage

Initial Login

1. Open your web browser and navigate to the application URL
2. Use the admin credentials you configured in the `.env` file
3. You'll be redirected to the dashboard upon successful login

Managing Subnets

1. Navigate to **Subnets** from the main menu

2. Click **Add Subnet** to create a new subnet
3. Fill in the required information:
   - **Subnet CIDR**: Network address with CIDR notation (e.g., 192.168.1.0/24)
   - **IP Pool Range**: Available IP range for dynamic assignment
   - **Default Gateway**: Router/Gateway IP address (optional)
   - **DNS Servers**: Comma-separated DNS server IP addresses (optional)
4. Click **Add Subnet** to save

Managing IP Reservations

1. Navigate to **Reservations** from the main menu
2. Click **Add Reservation** to create a new IP reservation
3. Fill in the required information:
   - **Subnet**: Choose the target subnet
   - **IP Address**: The IP address to reserve
   - **MAC Address**: MAC address of the device
   - **Hostname**: Device hostname (optional)
4. Click **Add Reservation** to save

Configuration Management

1. Navigate to **Configuration** from the main menu
2. Use the available actions:
   - **Update with Database Reservations**: Sync database reservations to Kea configuration
   - **Validate Configuration**: Check configuration for errors
   - **Reload Kea**: Apply configuration changes
   - **Create Backup**: Backup current configuration

Maintenance

Common Tasks

```
# Check application status
rake status

# Create configuration backup
rake backup_config

# Validate configuration
rake validate_config

# Clean old logs and backups
rake clean

# View all available tasks
rake help
```

Log Files

- **Application logs**: `logs/app.log`
- **Kea manager logs**: `logs/kea.log`
- **Database logs**: `logs/database.log`
- **Configuration logs**: `logs/config.log`

Backup Management

Configuration backups are automatically created in `/etc/kea/backups/` with timestamps. The application keeps the last 10 backups automatically.

Troubleshooting

Common Issues

1. **Database Connection Failed**
   - Check MySQL service status: `sudo systemctl status mysql`
   - Verify database credentials in `.env` file
   - Ensure database and user exist
2. **Kea Control Socket Not Found**
   - Check Kea configuration for control socket
   - Verify socket path in `.env` file
   - Ensure Kea service is running: `sudo systemctl status kea-dhcp4-server`
3. **Permission Denied Errors**
   - Ensure application user has read access to Kea configuration
   - Check file permissions on log directories
   - Verify MySQL user permissions
4. **Configuration Validation Fails**
   - Check Kea configuration syntax
   - Validate JSON format
   - Review Kea logs for detailed errors

Getting Help

1. Check the application logs for detailed error information
2. Verify system service status
3. Ensure all dependencies are properly installed
4. Check file permissions and ownership

Security Considerations

1. **Change Default Credentials**: Always change the default admin username and password

2. **Use HTTPS**: Configure SSL/TLS for production deployments
3. **Firewall**: Restrict access to the web interface
4. **Regular Updates**: Keep the system and dependencies updated
5. **Backup Strategy**: Implement regular configuration and database backups

Contributing

This application was created by MiniMax Agent. For issues or improvements, please check the application logs and system configuration.

License

This project is provided as-is for educational and practical use. Please ensure compliance with your organization's policies and applicable laws.

Kea DHCP Manager - Simplifying DHCP server management with modern web technologies.