# Test Plan

## 1. Introduction

This document describes the test plan for the Network Configuration Management System, covering tests for the ACL , Rule , Action , OutputAction , TunnelAction , Config , Interface , DP , Bgp , Router , Route , Vlan , and Vlans classes.

## 2. Test objectives

* Verify that all classes are initialized and properties are set correctly.
* Ensure that class methods can process and output data correctly.
* Examine interactions and dependencies between classes.
* Verify that the class can handle boundary conditions and erroneous data.

## 3. Testing strategy

* **Unit Testing** : Run unit tests on the constructors and main methods of each class to ensure they work as expected.
* **Integration testing** : Testing the interactions between classes to ensure they work together.
* **Boundary testing** : Checks a class's ability to handle extreme or invalid input.
* **Performance Testing** : Evaluate the performance of a system when processing large amounts of data.

## 4. Test Cases

### 4.1 ACL Class

**Use case 1** : Initialization test

* **Input** : ACL(name="test\_acl", rules=[{"rule": {"actset\_output": "output"}}])
* **Expected result** : The name attribute of the ACL object should be "test\_acl" , the rules attribute should contain a Rule object, and actions should be {"actset\_output": "output"} .

**Use case 2** : Property setting test

* **Input** : Rule(rule\_data={"dl\_dst": "00:11:22:33:44:55"})
* **Expected result** : dl\_dst attribute should be "00:11:22:33:44:55" .

### 4.2 Rule Class

**Use case 1** : Initialization test

* **Input** : Rule(rule\_data={"in\_port": 1, "actions": {"output": 2}})
* **Expected result** : the in\_port attribute should be 1 and the actions attribute should be {"output": 2} .

**Use case 2** : Default value testing

* **Input** : Rule(rule\_data={})
* **Expected result** : All attributes should be None or their default values.

### 4.3 Action Class

**Use case 1** : Initialization test

* **Input** : Action(action\_data={"allow": True, "cookie": "0x1234"})
* **Expected result** : allow attribute should be True and cookie attribute should be "0x1234" .

### 4.4 OutputAction Class

**Use case 1** : Initialization test

* **Input** : OutputAction(output\_data={"port": 1, "vlan\_vid": 100})
* **Expected result** : The port attribute should be 1 and the vlan\_vid attribute should be 100 .

### 4.5 TunnelAction Class

**Use case 1** : Initialization test

* **Input** : TunnelAction(tunnel\_data={"type": "vlan", "tunnel\_id": 10})
* **Expected result** : The type attribute should be "vlan" and the tunnel\_id attribute should be 10 .

### 4.6 Config Class

**Use case 1** : Initialization test

* **Input** : Config(vlans={}, routers=[], dps=[], acls=[])
* **Expected result** : vlans , routers , dps , and acls attributes should be empty dictionary, empty list, and empty list, respectively.

### 4.7 Interface Class

**Use case 1** : Initialization test

* **Input** : Interface(name="eth0", tagged\_vlans=[10], acls\_in=[1])
* **Expected results** : name attribute should be "eth0" , tagged\_vlans should be [10] , acls\_in should be [1] .

### 4.8 DP Class

**Use case 1** : Initialization test

* **Input** : DP(interfaces=[], dp\_id=1)
* **Expected result** : The dp\_id attribute should be 1 and interfaces should be an empty list.

### 4.9 Bgp Class

**Use case 1** : Initialization test

* **Input** : Bgp(vlan="vlan1", as\_number=65000)
* **Expected result** : The vlan attribute should be "vlan1" and the as\_number attribute should be 65000 .

### 4.10 Router Class

**Use case 1** : Initialization test

* **Input** : Router(vlans=[1], bgp=Bgp(vlan="vlan1", as\_number=65000))
* **Expected result** : vlans should be [1] , bgp should be a Bgp object with expected property values.

**Use case 2** : print\_details test

* **Input** : Call the print\_details method
* **Expected Result** : The output should match the details of the Router object.

### 4.11 Route Class

**Use case 1** : Initialization test

* **Input** : Route(ip\_dst="192.168.1.1", ip\_gw="192.168.1.254")
* **Expected result** : ip\_dst attribute should be "192.168.1.1" and ip\_gw attribute should be "192.168.1.254" .

### 4.12 Vlan Class

**Use case 1** : Initialization test

* **Input** : Vlan(vid=10, description="Test VLAN")
* **Expected result** : The vid attribute should be 10 and the description attribute should be "Test VLAN" .

### 4.13 Vlans Class

**Use case 1** : Initialization test

* **Input** : Vlans(vlans\_data={"vlan1": {"vid": 10}})
* **Expected Result** : The vlans attribute should contain a Vlan object of "vlan1" .

## 5. Test Environment

* **Operating System** : Linux / Windows
* **Python version** : Python 3.x
* **Dependency library** : No specific dependencies, standard library is sufficient
* **Testing tools** : pytest or unittest