# **SD-WAN Configuration Reports Playbook Documentation**

## **Overview**

The SD-WAN Configuration Reports playbook is an Ansible automation script designed to generate comprehensive configuration reports from Cisco SD-WAN environments. This playbook uses the Sastre tool to extract and analyze configuration data from the vManage controller, creating organized reports for different configuration types.

## **Playbook Details**

**File:** sastre\_report.yml  
 **Purpose:** Generate configuration reports using Sastre tool  
 **Target vManage Version:** 20.15  
 **Output Location:** generated/ folder (same directory as playbook)

## **Prerequisites**

* Ansible installed and configured
* Sastre tool installed and accessible in PATH
* Network connectivity to vManage controller
* Valid vManage credentials with read access
* Python environment with required SSL libraries

## **Environment Variables**

The playbook requires the following environment variables:

| **Variable** | **Description** | **Default Value** | **Required** |
| --- | --- | --- | --- |
| VMANAGE\_HOST | vManage controller hostname/IP | vmanage-amfament-prod.sdwan.cisco.com | Yes |
| VMANAGE\_USERNAME | vManage username | automation | Yes |
| VMANAGE\_PASSWORD | vManage password | None | Yes |

## **Directory Structure**

The playbook creates a simple directory structure:

* playbook\_directory/
* ├── sastre\_report.yml
* └── generated/
* ├── device\_templates\_[timestamp]/
* ├── feature\_templates\_[timestamp]/
* ├── policy\_definitions\_[timestamp]/
* ├── policy\_lists\_[timestamp]/
* ├── config\_groups\_[timestamp]/
* └── all\_items\_[timestamp]/

## **Task Analysis**

### **Task 1: Environment Variable Validation**

**Purpose:** Ensures all required credentials are available before proceeding

**What it does:**

* Validates that vmanage\_host, vmanage\_username, and vmanage\_password are set
* Fails the playbook immediately if any critical environment variables are missing
* Prevents failed report generation due to missing credentials

### **Task 2: Directory Creation**

**Purpose:** Creates the output directory for report storage

**Generated structure:**

* Creates generated/ folder in the same directory as the playbook
* Sets appropriate permissions (755)

### **Task 3: vManage Connectivity Testing**

**Purpose:** Verifies the vManage controller is accessible before attempting report generation

**What it does:**

* Makes a REST API call to /dataservice/system/device/controllers
* Uses basic authentication with provided credentials
* Sets 60-second timeout
* Ignores SSL certificate validation for internal certificates
* Stores results in connectivity\_test variable

### **Task 4: Connectivity Failure Handling**

**Purpose:** Stops execution if connectivity test fails

**What it does:** Prevents unnecessary report generation attempts when vManage is unreachable

### **Task 5: Generate Device Templates Report**

**Purpose:** Creates a comprehensive report of all device templates

**Generated content:**

* Device template configurations
* Template details and relationships
* Output saved to generated/device\_templates\_[timestamp]/

**Sastre parameters:**

* Report type: device\_template
* Verbose output enabled
* SSL verification disabled

### **Task 6: Generate Feature Templates Report**

**Purpose:** Creates a detailed report of all feature templates

**Generated content:**

* Feature template configurations
* Template parameters and values
* Output saved to generated/feature\_templates\_[timestamp]/

**Sastre parameters:**

* Report type: feature\_template
* Comprehensive template analysis

### **Task 7: Generate Policy Definitions Report**

**Purpose:** Analyzes and reports on all policy definitions

**Generated content:**

* Policy definition details
* Policy rules and configurations
* Output saved to generated/policy\_definitions\_[timestamp]/

**Sastre parameters:**

* Report type: policy\_definition
* Complete policy analysis

### **Task 8: Generate Policy Lists Report**

**Purpose:** Creates reports for all policy lists (data prefixes, sites, VPNs, etc.)

**Generated content:**

* Policy list configurations
* List entries and values
* Output saved to generated/policy\_lists\_[timestamp]/

**Sastre parameters:**

* Report type: policy\_list
* All policy list types included

### **Task 9: Generate Configuration Groups Report**

**Purpose:** Reports on configuration groups and their associations

**Generated content:**

* Configuration group details
* Device associations
* Output saved to generated/config\_groups\_[timestamp]/

**Sastre parameters:**

* Report type: config\_group
* Group membership analysis

### **Task 10: Generate Comprehensive All Items Report**

**Purpose:** Creates a complete report covering all configuration items

**Generated content:**

* All configuration types in one comprehensive report
* Cross-references between different configuration items
* Output saved to generated/all\_items\_[timestamp]/

**Sastre parameters:**

* Report type: all
* Complete environment analysis

## **Report Types Generated**

Each report generates detailed CSV files and analysis for:

1. **Device Templates Report**
   * Template names and IDs
   * Associated feature templates
   * Device attachments
   * Template status
2. **Feature Templates Report**
   * Template configurations
   * Parameter definitions
   * Usage across device templates
   * Template relationships
3. **Policy Definitions Report**
   * Policy rules and sequences
   * Match conditions
   * Actions and parameters
   * Policy status
4. **Policy Lists Report**
   * List types and entries
   * Data prefixes
   * Site lists
   * VPN lists
   * Application lists
5. **Configuration Groups Report**
   * Group configurations
   * Device memberships
   * Deployment status
   * Group hierarchies
6. **All Items Report**
   * Comprehensive overview
   * Cross-configuration relationships
   * Complete environment snapshot
   * Dependency mappings