PALO ALTO NETWORKS
SECURITY TECHNICAL IMPLEMENTATION GUIDE
(STIG)
ANSIBLE DOCUMENTATION

Version 1, Release 4

November 2021

Developed by DISA for the DoD

TABLE OF CONTENTS

Page

1. BACKGROUND

Ansible is an open source, cross-platform configuration management solution used to define and enforce system and application configurations. This package provides Ansible configurations that implement some of the Palo Alto Networks Network Device Management (NDM) Security Technical Implementation Guide (STIG). While the content has been tested during development, all possible system and environmental factors could not be tested. Before using this content in a production environment, please perform testing with the intended settings in your own test environment. There is no mandate to use this content; it is published as a resource to assist in the application of security guidance to your systems. Use it in the manner and to the extent that it assists with this goal.

2. INSTALLATION

The following instructions are for standalone installation using ansible-playbook[1] for testing purposes. A production environment may also use Ansible Tower. Refer here[2] for details.

2.1 Installing Ansible

Newer versions of Ansible are in the Red Hat Enterprise Linux 8 Extra Packages for Enterprise Linux (EPEL)[3] repository. To install it, run the following:

```
sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
sudo yum install ansible
```

For other installation methods, refer here[4].

2.2 Installing Required Python Packages

The Ansible **panos_config_element** module requires Python 3 and has some required Python packages to be installed on the host that executes the module. To install the requirements, run the following:

```
python3 -m pip install pan-os-python pandevice xmltodict
```

If pip is not installed, run the following:

```
sudo yum install python3-pip
python3 -m pip install --upgrade pip
```

2.3 Installing Required Ansible Collections

The Ansible **panos_config_element** module is part of the **paloaltonetworks.panos** Ansible collection. To use the collection, it must be installed from the Ansible Galaxy[5] via the following:

```
ansible-galaxy collection install paloaltonetworks.panos
```

For more information on installing the collection, refer here[6].

2.4 Extracting Content

Unzip the **panosSTIG-ansible.zip**.

[1] https://docs.ansible.com/ansible/latest/user_guide/playbooks_intro.html
[2] https://www.ansible.com/products/tower
[3] https://fedoraproject.org/wiki/EPEL
[4] https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#installation-guide
[5] https://galaxy.ansible.com/PaloAltoNetworks/panos
[6] https://paloaltonetworks.github.io/pan-os-ansible/#installation

3. CONFIGURATION

3.1 Simple

To apply the default STIG Ansible configuration, update the **`panosSTIG.yml`** file to enforce the STIG.

The **`panosSTIG.yml`** file is prepopulated with dummy values for **`ip_address, username,`** and **`password`**. Replace the values provided with the login details for the target Palo Alto system.

To store the username and password in an encrypted format, consider using Ansible Vault[7].

To apply the default STIG Ansible configuration, run the **`enforce.sh`** script to enforce the STIG on the target Palo Alto system.

To tailor the configuration, follow the steps in the next section.

3.2 Custom

To customize, create a YAML (.yml) file containing just the variables to customize from the variables named in the **`roles/panosSTIG/defaults/main.yml`** file. This file contains configuration data to define which configuration settings to manage and the values for these settings. Edit the newly created configuration file in a text editor to best suit each system's requirements as needed. For example, to turn off STIG rule ID 77249, set the "Manage" variable to **`False`**. To set STIG rule ID 77213's minimum password length to `20`, set the **`panosSTIG_stigrule_77213_password_minimum_length_Element`** variable to **`<minimum-length>20</minimum-length>`**.

```
panosSTIG_stigrule_77249_Manage: False
panosSTIG_stigrule_77249_timezone_Xpath:
'/config/devices/entry[@name="localhost.localdomain"]/deviceconfig/system'
panosSTIG_stigrule_77249_timezone_Element: '<timezone>GMT</timezone>'

panosSTIG_stigrule_77213_Manage: True
panosSTIG_stigrule_77213_password_minimum_length_Xpath: '/config/mgt-
config/password-complexity'
panosSTIG_stigrule_77213_password_minimum_length_Element:
'<enabled>yes</enabled>

<minimum-length>20</minimum-length>'
```

[7] https://docs.ansible.com/ansible/latest/user_guide/vault.html

To use the newly created custom variables file, edit **`site.yml`** to include it. See the highlighted line to add below:

```
- hosts: localhost
  Connection:local
  gather_facts: no
  collections:
    -  paloaltonetworks.panos
  vars_files:
    - panosSTIG.yml
    - /path/to/custom/vars.yml
  roles:
  - panosSTIG
```

For more information on variables, refer here[8]. For more information on YAML, refer here[9].

[8] https://docs.ansible.com/ansible/latest/user_guide/playbooks_variables.html
[9] https://docs.ansible.com/ansible/latest/reference_appendices/YAMLSyntax.html

4. COMPLIANCE EXTRACTION

This compliance extraction methodology returns results based on a system's compliance with the enforcement content. This may be different from STIG compliance. For example, multiple values may be allowed by the STIG but will be marked as "fail" if the value does not match the single exact value in the enforcement content. If a value is customized in a way that violates a STIG rule, it will be marked as "pass" because it matches the enforcement content's expected value.

Upon successful Ansible playbook play content extraction of the configuration results into XCCDF, results can be performed via an Ansible callback plugin. Use of this plugin can be controlled via modification of the following variable in the ansible.cfg file to include the name of the plugin to use:

```
[defaults]
callback_whitelist = stig_xml
```

Configuration of the plugin is controlled via creation/modification of the following environmental variables:

- **`export STIG_PATH=/path/to/stigs/stigs_are_here`**
- **`export XML_PATH=/path/where/to/write/results.xml`**

The above environmental variables control the plugin writing the XCCDF results to the file **`XML_PATH`** using the STIG at path **`STIG_PATH`**. The XCCDF results file is output by default to **`./xccdf-results.xml.`**

Note: The STIG provided above should match the STIG release and version number for which the Ansible content is built. A copy of the STIG is provided in **`roles/panosSTIG/files`**.

Ansible provides a means of checking compliance without enforcement called **`--check`** (aka "dry run"). To use this mode, run the following:

```
$ ansible-playbook -v -b -i /dev/null --check site.yml
```

5. OTHER CONSIDERATIONS

5.1 Saving Configuration

Additional functionality has been added to the playbook to enable committing the current configuration at the end of the play only if any of the tasks cause a configuration change. This functionality is disabled by default and can be enabled by setting the variable **`panosSTIG_save_configuration_Manage`** to **`True`**.