
ansible-role-apache Documentation

Release 1.10.3

Vladimir Botka

May 01, 2020

Contents:

1	User's guide	3
1.1	Introduction	4
1.2	Installation	4
1.3	Ansible playbook	4
1.4	Tags	5
1.5	Debug	6
1.6	Variables	6
2	Annotated source code	23
2.1	Tree	24
2.2	Variables	25
2.3	Packages	25
2.4	Configure	26
2.5	Service	41
2.6	Handlers	41
3	Examples	43
4	References	45
4.1	Requirements	45
4.2	Default variables	45
5	Indices and tables	47

Welcome to the Ansible role for Apache!

Topics

- *User's guide*
 - *Introduction*
 - *Installation*
 - *Ansible playbook*
 - *Tags*
 - *Debug*
 - *Variables*
 - * *Default variables*
 - * *OS specific default variables*
 - * *OS specific custom variables*
 - * *apache_vhost*
 - * *apache_directory_blocks*
 - * *apache_alias*
 - * *apache_conf_dir_vhosts*
 - * *apache_conf_dir_sections*
 - * *apache_httpd_conf*
 - * *apache_httpd_conf_ssl*
 - * *apache_httpd_conf_ssl_extra*
 - * *apache_httpd_conf_ssl_extra_absent*

```
* apache_httpd_conf_ssl_listen
* apache_httpd_conf_modules
```

Welcome to the User's Guide!

1.1 Introduction

The role install and configure Apache web server

- Ansible role: [Apache](#)
- Supported systems: [FreeBSD](#)
- Requirements:
 - [vbotka.ansible_lib](#)
 - [jtyr.config_encoder_filters](#)

The user of this role is expected to have read at least the following documents

- [Basic Concepts](#)
- [Roles](#)
- [Working With Playbooks](#)

1.2 Installation

The most convenient way how to install an Ansible role is to use Ansible Galaxy CLI `ansible-galaxy`. The utility comes with the standard Ansible packages and provide the user with simple interface to the Ansible Galaxy's services. For example take a look at the current status of the role

```
$ ansible-galaxy info vbotka.apache
```

and install it

```
$ ansible-galaxy install vbotka.apache
```

Together with the role `vbotka.apache` two other roles will be installed.

- [vbotka.ansible_lib](#) provides common tasks in the form of included tasks
- [jtyr.config_encoder_filters](#) provides the filter `encode_apache` used to encode YAML configuration data to Apache format.

See also:

- For details how to install specific versions from various sources see [Installing content](#).
- Take a look at other roles `$ ansible-galaxy search --author=vbotka`

1.3 Ansible playbook

Simple playbook to install and configure Apache at `srv.example.com` (2)


```

1 $ cat apache.yml
2 - hosts: srv.example.com
3   gather_facts: true
4   connection: ssh
5   remote_user: admin
6   become: yes
7   become_user: root
8   become_method: sudo
9   roles:
10  - vbotka.apache

```

Note:

- `gather_facts: true` (3) must be set to collect variables needed to evaluate *OS specific default variables* and *OS specific custom variables* [`ansible_distribution`, `ansible_distribution_release`, `ansible_os_family`]
- See [Variables](#)

See also:

- For details see [Connection Plugins](#) (4-5)
- and [Understanding Privilege Escalation](#) (6-8).

1.4 Tags

The tags provide very useful tool to run selected tasks of the role. To see what tags are available list the tags of the role with command below

```

1 $ ansible-playbook apache.yml --list-tags
2
3 playbook: apache.yml
4
5 play #1 (srv.example.conf): srv.example.com TAGS: []
6   TASK TAGS: [always, apache-debug, apache-httpd, apache-httpd-alias,
7     apache-httpd-confd, apache-httpd-confd-includes, apache-httpd-confd-vhosts,
8     apache-httpd-dirs, apache-httpd-modules, apache-httpd-ssl, apache-httpd-vhosts,
9     apache-packages, apache-service, apache-vars]

```

For example see the list of the variables and their values with the tag `apache-debug`

```
$ ansible-playbook apache.yml -t apache_debug -e 'apache_debug=true'
```

See what packages will be installed

```
$ ansible-playbook apache.yml -t apache_packages -e 'apache_debug=true' --check
```

Install packages only and exit the play. Enable the debug output

```
$ ansible-playbook apache.yml -t apache_packages -e 'apache_debug=true'
```

1.5 Debug

To see additional debug information in the output enable debug output in the configuration

```
apache_debug: true
```

, or set the extra variable in the command

```
$ ansible-playbook apache.yml -e 'apache_debug=true'
```

See also:

- [Playbook Debugger](#)

1.6 Variables

In this guide we describe role defaults variables in the directory `defaults` and variables included from the directory `vars`.

- role defaults in the directory `{{ role_path }}/defaults` (precedence 2.)
- include OS specific vars from the directory `{{ role_path }}/vars` (precedence 18.)

See also:

[Ansible variable precedence: Where should I put a variable?](#)

1.6.1 Default variables

- Most of the variables are self-explaining (4-9,12-13,64-65)
- For Apache configuration (19-54,60) see [Apache HTTP Server Documentation](#).
- Other variables (70,73,76,79-80) will be explained in the next sections.

[defaults/main.yml]

```

1  ---
2  # defaults for vbotka.apache
3
4  apache_install: true
5  apache_enable: true
6  apache_debug: false
7  apache_ssl: false
8  apache_php: false
9  apache_backup_conf: false
10
11  apache_SSEngine: "off"
12
13  # httpd.conf
14  apache_ServerName: "www.example.com"
15  apache_ServerAdmin: "admin@example.com"
16  apache_httpd_conf:
17  - {regexp: "ServerName", line: "{{ apache_ServerName }}" }
18  - {regexp: "ServerAdmin", line: "{{ apache_ServerAdmin }}" }
19

```

(continues on next page)

(continued from previous page)

```

20 # SSL
21 apache_SSLListen: ""
22 apache_SSLCertificateFile: "/usr/local/etc/apache{{ apache_version }}/server.crt"
23 apache_SSLCertificateKeyFile: "/usr/local/etc/apache{{ apache_version }}/server.key"
24 apache_SSLProtocol: "all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1"
25 # apache_SSLEngine: HIGH:!aNULL:!MD5
26 # apache_SSLEngine: RC4-SHA:AES128-SHA:HIGH:!aNULL:!MD5
27 apache_SSLEngine: "ECDHE-ECDSA-AES256-GCM-SHA384:\
28 ECDHE-RSA-AES256-GCM-SHA384:\
29 ECDHE-ECDSA-CHACHA20-POLY1305:\
30 ECDHE-RSA-CHACHA20-POLY1305:\
31 ECDHE-ECDSA-AES128-GCM-SHA256:\
32 ECDHE-RSA-AES128-GCM-SHA256:\
33 ECDHE-ECDSA-AES256-SHA384:\
34 ECDHE-RSA-AES256-SHA384:\
35 ECDHE-ECDSA-AES128-SHA256:\
36 ECDHE-RSA-AES128-SHA256"
37 apache_SSLHonorCipherOrder: "on"
38 apache_SSLCompression: "off"
39 apache_SSLSessionTickets: "off"
40 # SSLOpenSSLConfCmd DHParameters "/usr/local/etc/ssl/dhparam.pem"
41 # SSLSessionCache shmcb:/var/run/ssl_scache(512000)
42 # Header always set X-Frame-Options DENY
43 # Header always set X-Frame-Options SAMEORIGIN
44 apache_httpd_conf_ssl:
45 - "Include etc/apache{{ apache_version }}/extra/httpd-ssl.conf"
46 apache_httpd_conf_ssl_extra:
47 - {regexp: "ServerName ", line: "{{ apache_ServerName }}:443"}
48 - {regexp: "ServerAdmin ", line: "{{ apache_ServerAdmin }}" }
49 - {regexp: "SSLEngine ", line: "{{ apache_SSLEngine }}" }
50 - {regexp: "SSLProtocol ", line: "{{ apache_SSLProtocol }}" }
51 - {regexp: "SSLEngine ", line: "{{ apache_SSLEngine }}" }
52 - {regexp: "SSLHonorCipherOrder ", line: "{{ apache_SSLHonorCipherOrder }}" }
53 - {regexp: "SSLCompression ", line: "{{ apache_SSLCompression }}" }
54 - {regexp: "SSLSessionTickets ", line: "{{ apache_SSLSessionTickets }}" }
55 - {regexp: "SSLCertificateFile ", line: "{{ apache_SSLCertificateFile }}" }
56 - {regexp: "SSLCertificateKeyFile ", line: "{{ apache_SSLCertificateKeyFile }}" }
57 apache_httpd_conf_ssl_extra_absent: []
58 apache_httpd_conf_ssl_listen:
59 - "Listen 443"
60
61 # Modules
62 apache_httpd_conf_modules:
63 - {module: "socache_shmcb_module", mod: "mod_socache_shmcb.so"}
64
65 # PHP
66 apache_php_version: "56"
67 apache_php_package: "www/mod_php{{ apache_php_version }}"
68
69 # vhosts
70 # Virtual hosts need apache_ssl. Port 80 is redirected permanently to
71 # 443 for vhosts.
72 apache_vhost: []
73
74 # dirs
75 apache_directory_blocks: []
76

```

(continues on next page)

(continued from previous page)

```

77 # aliases
78 apache_alias: []
79
80 # conf.d
81 apache_confdir_vhosts: "{{ role_path }}/vars/conf.d/vhosts"
82 apache_confdir_sections: "{{ role_path }}/vars/conf.d/sections"
83
84 # EOF
85 ...

```

Warning: By default SSL is turned off `apache_SSLEngine: "off" (9)`.

1.6.2 OS specific default variables

Here come the OS specific default variables. The configuration files in the directory `vars/defaults` will be included with `with_first_found(1)`. At least empty `default.yml (6)` shall be present.

- `[tasks/vars.yml]`
- `[al_include_os_vars_path.yml]`

```

1 with_first_found:
2 - files:
3   - "{{ ansible_distribution }}-{{ ansible_distribution_release }}.yml"
4   - "{{ ansible_distribution }}.yml"
5   - "{{ ansible_os_family }}.yml"
6   - "default.yml"
7   - "defaults.yml"
8 paths: "{{ al_os_vars_path }}/vars/defaults"

```

Note:

- OS specific variables are included with the module `include_var` that has very high precedence (18 in the list of 22).
- See [Ansible variable precedence: Where should I put a variable?](#)
- To override the default variables see *OS specific custom variables*

FreeBSD default variables

By default the binary packages will be installed (4). But if custom builds are available switch to `ports (5)` and use `freebsd_use_packages: "yes" (6)` to speedup the installation. Under standard circumstances, there is no reason to change other parameters here.

`[vars/defaults/FreeBSD.yml]`

```

1 ---
2 # FreeBSD defaults for vbotka.apache
3
4 freebsd_install_method: "packages"
5 # freebsd_install_method: "ports"

```

(continues on next page)

(continued from previous page)

```

6  freebsd_use_packages: "yes"
7  freebsd_install_retries: 10
8  freebsd_install_delay: 5
9
10 apache_version: "24"
11 apache_package: "www/apache{{ apache_version }}"
12 apache_dir: "apache{{ apache_version }}"
13 apache_service: "apache{{ apache_version }}"
14 apache_conf_path: "/usr/local/etc/apache{{ apache_version }}"
15 apache_data_owner: "www"
16 apache_data_group: "wheel"
17 apache_data_mode: "0640"
18 apache_dir_mode: "0750"
19 apache_packages:
20   - "{{ apache_package }}"
21
22 # EOF
23 ...

```

1.6.3 OS specific custom variables

Here come the OS specific custom variables. The configuration files in the directory `vars` will be included with `with_first_found` (1) and will override the default values of the variables. At least empty `default.yml` (6) shall be present here.

- `[tasks/vars.yml]`
- `[al_include_os_vars_path.yml]`

```

1  with_first_found:
2  - files:
3    - "{{ ansible_distribution }}-{{ ansible_distribution_release }}.yml"
4    - "{{ ansible_distribution }}.yml"
5    - "{{ ansible_os_family }}.yml"
6    - "default.yml"
7    - "defaults.yml"
8  paths: "{{ al_os_vars_path }}/vars"

```

Note:

- OS specific variables from the directory `{{ al_os_vars_path }}/vars` override OS specific default variables from the directory `{{ al_os_vars_path }}/vars/defaults`.
- See `al_include_os_vars_path.yml`.

1.6.4 apache_vhost

- *Synopsis*
- *Parameters*
- *Example*

- *Notes*
- *See Also*

Synopsis

- `apache_vhost` is a list of virtual hosts.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
ServerName	<i>string</i> required	Fully qualified domain name (FQDN)
DocumentRoot	<i>string</i> required	Path DocumentRoot
SSLCertificateFile	<i>string</i> required	Path to SSL Certificate
SSLCertificateKeyFile	<i>string</i> required	Path to SSL Private key
redirect	<i>boolean</i> default: <code>false</code>	Redirect permanent http to https
create_document_root	<i>boolean</i> default: <code>false</code>	Create DocumentRoot

Example

The example below will configure virtual server `mail.example.net` (2).

```

1 apache_vhost :
2   - ServerName: "mail.example.net"
3     DocumentRoot: "/usr/local/www/roundcube/"

```

(continues on next page)

(continued from previous page)

```
4  SSLCertificateFile: "/usr/local/etc/letsencrypt/live/mail.example.net/fullchain.  
↪pem"  
5  SSLCertificateKeyFile: "/usr/local/etc/letsencrypt/live/mail.example.net/privkey.  
↪pem"
```

Notes

Note:

- The default value is an empty dictionary `apache_vhost: {}`
- For details see *Configure vhosts*. [[httpd-vhosts.yml](#)]

See Also

See also:

- It is also possible to configure virtual servers with `apache_conf_dir_vhosts`. See *apache_conf_dir_vhosts*.

1.6.5 apache_directory_blocks

- *Synopsis*
- *Parameters*
- *Example*
- *Notes*
- *See Also*

Synopsis

- `apache_directory_blocks` is a list of directory blocks.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
Directory	<i>string</i> required	DocumentRoot directory
Includefile	<i>string</i> required	Path to the includefile to be created
Conf	<i>list</i>	Configuration of the directory

Example

Configuration file (3) will be created in the directory `{{ apache_conf_path }}/Includes/`.

```

1  apache_directory_blocks:
2    - Directory: "/usr/local/www/roundcube"
3    Includefile: "usr-local-www-roundcube.conf"
4    Conf:
5      - "Options Indexes FollowSymLinks"
6      - "DirectoryIndex index.html"
7      - "AllowOverride All"
8      - "Require all granted"

```

Notes

Note:

- The default value is an empty dictionary `apache_directory_blocks: {}`
 - For details see *Configure directories in Includes*. [[httpd-dirs.yml](#)]
-

See Also

See also:

- *Template directory-block.j2*

1.6.6 apache_alias

- *Synopsis*
- *Example*
- *Notes*

Synopsis

- `apache_alias` is a list of aliases.

Example

```
1 apache_alias:  
2 - "ScriptAlias /nagios/cgi-bin/ /usr/local/www/nagios/cgi-bin/"  
3 - "Alias /nagios/ /usr/local/www/nagios/"  
4 - "Alias /joomla /usr/local/www/joomla3/"
```

Notes

Note:

- The default value is an empty list `apache_alias: []`
- For details see *Configure alias*. [[httpd-alias.yml](#)]

1.6.7 apache_confdir_vhosts

- *Synopsis*
- *Parameters*
- *Example*
- *Notes*
- *Hints*

Synopsis

- `apache_confdir_vhosts` is path to directory with virtual hosts' configuration files.

Parameters

The parameters and format of the files are described in the filter `encode_apache`.

Example

From the configuration file below the configuration file `{{ apache_conf_path }}/extra/mail.example.net.conf` will be created and the file will be included in `{{ apache_conf_path }}/httpd.conf`.

```
1 $ cat mail.example.net/apache.d/vhosts/mail.example.net.yml
2 my_apache_vhost:
3   content:
4     - sections:
5       - name: VirtualHost
6         param: "*:80"
7         content:
8           - options:
9             - ServerName: mail.example.net
10            - DocumentRoot: /usr/local/www/roundcube/
11            - Redirect permanent /: https://mail.example.net/
12      - sections:
13        - name: VirtualHost
14          param: "*:443"
15          content:
16            - options:
17              - ServerName: mail.example.net
18              - DocumentRoot: /usr/local/www/roundcube/
19              - SSLCertificateFile: /usr/local/etc/ssl/certs/mail.example.net.crt
20              - SSLCertificateKeyFile: /usr/local/etc/ssl/private/mail.example.net.
↵key
```

Notes

Note:

- For details see *Configure confd-vhosts*. [[httpd-confd-vhosts.yml](#)]
-

Hints

Hint:

- The default value is
`apache_conf_dir_vhosts: "{{ role_path }}/vars/conf.d/vhosts"`
 - In projects it might be convenient to change the path. For example
`apache_conf_dir_vhosts: "{{ playbook_dir }}/apache.d/vhosts"`
-

1.6.8 apache_conf_dir_sections

- *Synopsis*
- *Parameters*

- *Example*
- *Notes*
- *Hints*

Synopsis

- `apache_conf_dir_sections` is path to directory with configuration files.

Parameters

The parameters and format of the files are described in the filter `encode_apache`. The content of the files will be encoded and stored in the files in the directory `{{ apache_conf_path }}/Includes/`.

Example

For example from the configuration file below the configuration file `usr-local-ww-rouncube.conf` will be created and stored in the directory `{{ apache_conf_path }}/Includes (17)`.

```

1 $ cat mail.example.net/apache.d/sections/usr-local-ww-rouncube.yml
2 my_apache_dir:
3   content:
4     - sections:
5       - name: Directory
6         param: /usr/local/www/rouncube
7         content:
8           - options:
9             - Options:
10              - Indexes
11              - FollowSymLinks
12             - AllowOverride: All
13             - Require:
14              - all
15              - granted
16
17 $ cat /usr/local/etc/apache24/Includes/usr-local-ww-rouncube.conf
18 <Directory /usr/local/www/rouncube>
19   Options Indexes FollowSymLinks
20   AllowOverride All
21   Require all granted
22 </Directory>

```

Notes

Note:

- For details see *Configure confd-includes*. [[httpd-confd-includes.yml](#)]

Hints

Hint:

- The default value is
`apache_conf_dir_vhosts: "{{ role_path }}/vars/conf.d/sections"`
 - In projects it might be convenient to change the path. For example
`apache_conf_dir_vhosts: "{{ playbook_dir }}/apache.d/sections"`
-

1.6.9 apache_httpd_conf

- *Synopsis*
- *Parameters*
- *Example*
- *Notes*

Synopsis

- `apache_httpd_conf` is a list of lines in `httpd.conf`.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
regex	<i>string</i> required	The pattern to replace if found
line	<i>string</i> required	The line to insert/replace into the file

Example

```

1 apache_httpd_conf:
2   - {regex: "ServerName", line: "{{ apache_ServerName }}" }
3   - {regex: "ServerAdmin", line: "{{ apache_ServerAdmin }}" }
4   - {regex: "ServerRoot", line: "/usr/local"}
5   - {regex: "MIMEMagicFile", line: "etc/apache24/magic"}

```

Notes

Note:

* The default value is

```
apache_httpd_conf:
  - {regexp: "ServerName", line: "{{ apache_ServerName }}"}
```

* The argument line must be quoted if it contains spaces

```
- {regexp: "ErrorDocument 500", line: "\"The server made a boo boo.\""}
```

* For details see [Configure httpd.conf](#). [httpd.yml]

1.6.10 apache_httpd_conf_ssl

- [Synopsis](#)
- [Parameters](#)
- [Notes](#)

Synopsis

- `apache_httpd_conf_ssl` is a list of lines that configure SSL in `httpd.conf`.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
line	<i>string</i> required	The line to insert into the file

Notes

Note:

* The default value is

```
apache_httpd_conf_ssl:
  - "Include etc/apache{{ apache_version }}/extra/httpd-ssl.conf"
```

* For details see [Configure ssl](#). [httpd-ssl.yml]

1.6.11 apache_httpd_conf_ssl_extra

- *Synopsis*
- *Parameters*
- *Notes*

Synopsis

- `apache_httpd_conf_ssl_extra` is a list of lines that configure SSL in `extra/httpd-ssl.conf`.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
regexp	<i>string</i> required	The pattern to replace if found
line	<i>string</i> required	The line to insert/replace into the file

Notes

Note:

- See the default value in *Default variables*
 - For details see *Configure ssl*. [[httpd-ssl.yml](#)]
-

1.6.12 apache_httpd_conf_ssl_extra_absent

- *Synopsis*
- *Parameters*
- *Notes*

Synopsis

- `apache_httpd_conf_ssl_extra_absent` is a list of lines that will be removed from `extra/httpd-ssl.conf`.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
regexp	<i>string</i> required	The pattern to be removed

Notes

Note:

- The default value is empty list `apache_httpd_conf_ssl_extra_absent: []`
- For details see *Configure ssl*. [[httpd-ssl.yml](#)]

1.6.13 apache_httpd_conf_ssl_listen

- *Synopsis*
- *Notes*

Synopsis

- `apache_httpd_conf_ssl_listen` is a list of addresses and ports that the server will bind to.

Notes

Note:

- The default value is


```
apache_httpd_conf_ssl_listen:
  - "Listen 443"
```
- Overlapping Listen directives will result in a fatal error which will prevent the server from starting up.
- For details see *Configure ssl*. [[httpd-ssl.yml](#)]

1.6.14 apache_httpd_conf_modules

- *Synopsis*
- *Parameters*
- *Example*
- *Notes*

Synopsis

- `apache_httpd_conf_modules` is a list of modules to be loaded.

Parameters

<i>Parameter</i>	<i>Type</i>	<i>Comments</i>
module	<i>string</i> required	Name of the module
mod	<i>string</i> required	Object file or Library
present	<i>boolean</i> default: true	If <code>true</code> <code>LoadModule</code> directive will be added to <code>httpd.conf</code> . If <code>false</code> directive will be commented (disabled).

Example

```

1 apache_httpd_conf_modules:
2   - {module: "socache_shmcb_module", mod: "mod_socache_shmcb.so"}
3   - {module: "ssl_module", mod: "mod_ssl.so"}
4   - {module: "php5_module", mod: "libphp5.so"}

```


Notes

Note:

- The default value is

```
apache_httpd_conf_modules:  
  - {module: "socache_shmcb_module", mod: "mod_socache_shmcb.so"}
```
 - For details see *Configure modules*. [httpd-modules.yml]
-

Topics

- *Annotated source code*
 - *Tree*
 - *Variables*
 - * *Include OS specific variables*
 - *Packages*
 - * *Install packages*
 - * *Install FreeBSD packages*
 - *Configure*
 - * *Configure httpd.conf*
 - * *Configure directories in Includes*
 - * *Configure modules*
 - * *Configure alias*
 - * *Configure ssl*
 - * *Configure vhosts*
 - * *Configure confd*
 - * *Configure confd-vhosts*
 - * *Configure confd-includes*
 - *Service*
 - *Handlers*

2.1 Tree

```
$ tree .
.
├── defaults
│   └── main.yml
├── handlers
│   └── main.yml
├── LICENSE
├── meta
│   └── main.yml
├── README.md
├── requirements.yml
├── tasks
│   ├── debug.yml
│   ├── fn
│   │   └── httpd-confd-vhost-dirs.yml
│   ├── httpd-alias.yml
│   ├── httpd-confd-includes.yml
│   ├── httpd-confd-vhosts.yml
│   ├── httpd-confd.yml
│   ├── httpd-dirs.yml
│   ├── httpd-modules.yml
│   ├── httpd-ssl.yml
│   ├── httpd-vhosts.yml
│   ├── httpd.yml
│   ├── main.yml
│   ├── packages-freebsd.yml
│   ├── packages.yml
│   ├── rcconf.yml
│   ├── service.yml
│   └── vars.yml
├── templates
│   ├── directory-block.j2
│   ├── section2.j2
│   ├── vhost2.j2
│   └── vhost.j2
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    ├── conf.d
    │   ├── sections
    │   ├── sections-sample
    │   │   └── usr-local-www-example.com.yml
    │   ├── vhosts
    │   └── vhosts-sample
    │       └── example.com.yml
    ├── defaults
    │   ├── default.yml
    │   └── FreeBSD.yml
    ├── default.yml
    └── main.yml
```

2.2 Variables

In this guide we describe role defaults variables in the directory `defaults` and variables included from the directory `vars`.

- role defaults in the directory `{{ role_path }}/defaults` (precedence 2.)
- include OS specific vars from the directory `{{ role_path }}/vars` (precedence 18.)

See also:

Variable precedence: Where should I put a variable?

2.2.1 Include OS specific variables

Synopsis: Include OS specific variables from the role's directory `vars`.

OS specific default variables will be loaded from the files in the directory `vars/defaults`. OS specific custom variables, that will override default values, can be loaded from the files in the directory `vars`.

[vars.yml]

```

1 ---
2
3 - name: "vars: Include OS vars"
4   block:
5     - name: "vars: Set variables for al_include_os_vars_path"
6       set_fact:
7         al_os_vars_path: "{{ role_path }}"
8     - name: "vars: Include OS vars"
9       include_role:
10        name: "vbotka.ansible_lib"
11        tasks_from: "al_include_os_vars_path.yml"
12
13 # EOF
14 ...

```

See also:

- See the precedence, naming conventions and other details in the included tasks (11) from `al_include_os_vars_path.yml`.
- See *Variables*.

2.3 Packages

2.3.1 Install packages

Synopsis: Install packages for supported OS.

[packages.yml]

```

1 ---
2
3 - name: "packages: Install FreeBSD packages"
4   import_tasks: "packages-freebsd.yml"

```

(continues on next page)

(continued from previous page)

```

5  when: ansible_os_family == "FreeBSD"
6
7  # EOF
8  ...

```

2.3.2 Install FreeBSD packages

[packages-freebsd.yml]

```

1  ---
2
3  - name: "packages: Install packages"
4    when: freebsd_install_method == "packages"
5    pkgng:
6      name: "{{ item }}"
7      loop: "{{ apache_packages }}"
8      register: result
9      until: result is succeeded
10     retries: "{{ freebsd_install_retries }}"
11     delay: "{{ freebsd_install_delay }}"
12
13  - name: "packages: Install ports"
14    when: freebsd_install_method == "ports"
15    portinstall:
16      name: "{{ item }}"
17      use_packages: "{{ freebsd_use_packages|default(true) }}"
18      loop: "{{ apache_packages }}"
19      register: result
20      until: result is succeeded
21      retries: "{{ freebsd_install_retries }}"
22      delay: "{{ freebsd_install_delay }}"
23
24  # EOF
25  ...

```

2.4 Configure

Topics

- *Annotated source code*
 - *Tree*
 - *Variables*
 - * *Include OS specific variables*
 - *Packages*
 - * *Install packages*
 - * *Install FreeBSD packages*
 - *Configure*

- * *Configure httpd.conf*
 - *Synopsis*
 - *Annotation*
 - *Code*
 - *See Also*
- * *Configure directories in Includes*
 - *Synopsis*
 - *Annotation*
 - *Code*
 - *Template directory-block.j2*
 - *See Also*
- * *Configure modules*
 - *Synopsis*
 - *Annotation*
 - *Code*
 - *See Also*
- * *Configure alias*
 - *Synopsis*
 - *Annotation*
 - *Code*
- * *Configure ssl*
 - *Synopsis*
 - *Annotation*
 - *Code*
 - *See Also*
- * *Configure vhosts*
 - *Synopsis*
 - *Annotation*
 - *Code*
 - *Template vhost.j2*
 - *See Also*
- * *Configure confd*
 - *Synopsis*
 - *Annotation*
 - *Code*

- * *Configure confd-vhosts*
 - *Synopsis*
 - *Annotation*
 - *Include data from conf.d (3-17)*
 - *Create directories for virtual hosts (25-27)*
 - *Configure virtual hosts in extra directory (29-38)*
 - *Include virtual hosts in httpd.conf (40-48)*
 - *Configure confd-vhost-dirs*
 - * *Configure confd-includes*
 - *Synopsis*
 - *Annotation*
 - *Include data from conf.d (3-17)*
 - *Configure sections in Includes directory (19-28)*
 - *Template section2.j2*
- *Service*
- *Handlers*

2.4.1 Configure httpd.conf

- *Synopsis*
- *Annotation*
- *Code*
- *See Also*

Synopsis

- Configure lines in httpd.conf

Annotation

Iterate the list `{{ apache_httpd_conf }}` (9) and add lines to `{{ apache_conf_path }}/httpd.conf` (5).

Code

[httpd.yml]


```

1 ---
2
3 - name: "httpd_conf: Configure parameters in {{ apache_conf_path }}/httpd.conf"
4   lineinfile:
5     dest: "{{ apache_conf_path }}/httpd.conf"
6     regexp: "^{{ item.regexp }}\s"
7     line: "{{ item.regexp }} {{ item.line }}"
8     backup: "{{ apache_backup_conf }}"
9     loop: "{{ apache_httpd_conf }}"
10    notify: reload apache
11
12 # EOF
13 ...

```

See Also

See also:

Variable `apache_httpd_conf`

2.4.2 Configure directories in Includes

- *Synopsis*
- *Annotation*
- *Code*
 - *Template directory-block.j2*
- *See Also*

Synopsis

- Create files with the directory blocks in the Includes directory.

Annotation

Iterate the list `{{ apache_directory_blocks }}` (11) and create configuration files in the directory `{{ apache_conf_path }}/Includes` (6).

Code

[httpd-dirs.yml]

```

1 ---
2
3 - name: "httpd-dirs: Configure directories in {{ apache_conf_path }}/Includes/"
4   template:
5     src: "directory-block.j2"

```

(continues on next page)

(continued from previous page)

```

6  dest: "{{ apache_conf_path }}/Includes/{{ item.Includefile }}"
7  owner: "root"
8  group: "www"
9  mode: "0644"
10 backup: "{{ apache_backup_conf }}"
11 loop: "{{ apache_directory_blocks }}"
12 notify: reload apache
13
14 # EOF
15 ...

```

Template directory-block.j2

[directory-block.j2]

```

1 <Directory {{ item.Directory }}>
2 {% for confitem in item.Conf %}
3   {{ confitem }}
4 {% endfor %}
5 </Directory>

```

See Also

See also:

Variable *apache_directory_blocks*

2.4.3 Configure modules

- *Synopsis*
- *Annotation*
- *Code*
- *See Also*

Synopsis

- Load Apache modules. Optionally configure PHP module.
- (TODO: General configuration of modules.)

Annotation

- Iterate `apache_httpd_conf_modules` (11). When `item.preset` (4) insert line `LoadModule ...` (8) to `httpd.conf` (6)
- Iterate `apache_httpd_conf_modules` (22). When not `item.preset` (15) comment line `# LoadModule ...` (20) in `httpd.conf` (17)

- Configure PHP (30-38) in Includes/php.conf when apache_php (26)

Code

[httpd-modules.yml]

```

1  ---
2
3  - name: "httpd-modules: Load modules in {{ apache_conf_path }}/httpd.conf"
4    when: item.present|default(true)
5    lineinfile:
6      dest: "{{ apache_conf_path }}/httpd.conf"
7      regexp: "^\\s*#*\\s*LoadModule {{ item.module }}"
8      line: "    LoadModule {{ item.module }} libexec/{{ apache_dir }}/{{ item.mod }}"
9      insertbefore: "LoadModule"
10     backup: "{{ apache_backup_conf }}"
11     loop: "{{ apache_httpd_conf_modules }}"
12     notify: restart apache
13
14  - name: "httpd-modules: Unload modules in {{ apache_conf_path }}/httpd.conf"
15    when: not item.present|default(true)
16    lineinfile:
17      dest: "{{ apache_conf_path }}/httpd.conf"
18      regexp: "^\\s*#*\\s*LoadModule {{ item.module }}"
19      line: "    # LoadModule {{ item.module }}"
20      insertbefore: "LoadModule"
21      backup: "{{ apache_backup_conf }}"
22     loop: "{{ apache_httpd_conf_modules }}"
23     notify: restart apache
24
25  - name: "httpd-modules: Config PHP in {{ apache_conf_path }}/Includes/php.conf"
26    when: apache_php|bool
27    blockinfile:
28      dest: "{{ apache_conf_path }}/Includes/php.conf"
29      block: |
30        <IfModule dir_module>
31          DirectoryIndex index.html index.php
32        </IfModule>
33        <FilesMatch "\.php$">
34          SetHandler application/x-httpd-php
35        </FilesMatch>
36        <FilesMatch "\.phps$">
37          SetHandler application/x-httpd-php-source
38        </FilesMatch>
39        owner: "root"
40        group: "www"
41        mode: "0640"
42        create: "yes"
43        backup: "{{ apache_backup_conf }}"
44    notify: restart apache
45
46  # EOF
47  ...

```

See Also

See also:

Variable *apache_httpd_conf_modules*

2.4.4 Configure alias

- *Synopsis*
- *Annotation*
- *Code*

Synopsis

- Configure aliases in httpd.conf

Annotation

- When not an empty list (4) iterate *apache_alias* (9) and include configuration lines in httpd/conf (6)

Code

[httpd-alias.yml]

```

1  ---
2
3  - name: "httpd-alias: Configure aliases in {{ apache_conf_path }}/httpd.conf"
4    when: apache_alias|length > 0
5    blockinfile:
6      dest: "{{ apache_conf_path }}/httpd.conf"
7      insertafter: "<IfModule alias_module>"
8      block: |2
9          {% for item in apache_alias %}
10             {{ item }}
11          {% endfor %}
12      backup: "{{ apache_backup_conf }}"
13      notify: reload apache
14
15 # EOF
16 ...

```

See also:

Variable *apache_alias*

2.4.5 Configure ssl

- *Synopsis*
- *Annotation*
- *Code*
- *See Also*

Synopsis

- Configure SSL in extra/httpd-ssl.conf

Annotation

- Iterate `apache_httpd_conf_ssl_extra` (12) and configure lines in `extra/httpd-ssl.conf`
- Iterate `apache_httpd_conf_ssl_extra_absent` (20) and remove lines from `extra/httpd-ssl.conf` (17)
- Iterate `apache_httpd_conf_ssl_listen` (27) and add lines from `extra/httpd-ssl.conf` (17)
- Iterate `apache_httpd_conf_ssl` (34) and configure lines in `httpd`

Code

[httpd-ssl.yml]

```

1  ---
2
3  - name: "httpd-ssl: Configure {{ apache_conf_path }}/extra/httpd-ssl.conf"
4    when: apache_ssl|bool
5    block:
6      - name: "httpd-ssl: Present extra lines in {{ apache_conf_path }}/extra/httpd-ssl.
7        ↪conf"
8        lineinfile:
9          dest: "{{ apache_conf_path }}/extra/httpd-ssl.conf"
10         regexp: "^{{ item.regexp }}"
11         line: "{{ item.regexp }}{{ item.line }}"
12         backup: "{{ apache_backup_conf }}"
13         loop: "{{ apache_httpd_conf_ssl_extra }}"
14         notify: reload apache
15      - name: "httpd-ssl: Absent extra lines in {{ apache_conf_path }}/extra/httpd-ssl.
16        ↪conf"
17        lineinfile:
18         state: "absent"
19         dest: "{{ apache_conf_path }}/extra/httpd-ssl.conf"
20         regexp: "{{ item }}"
21         backup: "{{ apache_backup_conf }}"
22         loop: "{{ apache_httpd_conf_ssl_extra_absent }}"
23         notify: reload apache
24      - name: "httpd-ssl: SSL Listen in {{ apache_conf_path }}/extra/httpd-ssl.conf"
25        lineinfile:
26         dest: "{{ apache_conf_path }}/extra/httpd-ssl.conf"
27         line: "{{ item }}"
28         backup: "{{ apache_backup_conf }}"

```

(continues on next page)

(continued from previous page)

```

27     loop: "{{ apache_httpd_conf_ssl_listen }}"
28     notify: reload apache
29     - name: "httpd-ssl: SSL in {{ apache_conf_path }}/httpd.conf"
30       lineinfile:
31         dest: "{{ apache_conf_path }}/httpd.conf"
32         line: "{{ item }}"
33         backup: "{{ apache_backup_conf }}"
34         loop: "{{ apache_httpd_conf_ssl }}"
35         notify: reload apache
36
37 # EOF
38 ...

```

See Also

See also:

- Variable *apache_httpd_conf_ssl*
- Variable *apache_httpd_conf_ssl_extra*
- Variable *apache_httpd_conf_ssl_extra_absent*
- Variable *apache_httpd_conf_ssl_listen*

2.4.6 Configure vhosts

- *Synopsis*
- *Annotation*
- *Code*
 - *Template vhost.j2*
- *See Also*

Synopsis

- Configure virtual hosts in extra directory.

Annotation

- Loop the dictionary `{{ apache_vhost }}` (10,21,31) and
- optionally (11) create directories `DocumentRoot` (5,6).
- create files with the Apache virtual hosts in `{{ apache_conf_path }}/extra/` (16). See *Template vhost.j2* (15).
- Include created files (28) in `{{ apache_conf_path }}/httpd.conf` (26).

Code

[httpd-vhosts.yml]

```

1 ---
2
3 - name: "httpd-vhosts: Create directories for virtual hosts"
4   file:
5     state: "directory"
6     path: "{{ item.DocumentRoot }}"
7     owner: "{{ apache_data_owner }}"
8     group: "{{ apache_data_group }}"
9     mode: "{{ apache_dir_mode }}"
10    loop: "{{ apache_vhost }}"
11    when: item.create_document_root|default(False)
12
13 - name: "httpd-vhosts: Configure virtual hosts in {{ apache_conf_path }}/extra/"
14   template:
15     src: "vhost.j2"
16     dest: "{{ apache_conf_path }}/extra/{{ item.ServerName }}.conf"
17     owner: "{{ apache_data_owner }}"
18     group: "{{ apache_data_group }}"
19     mode: "{{ apache_data_mode }}"
20     backup: "{{ apache_backup_conf }}"
21     loop: "{{ apache_vhost }}"
22     notify: reload apache
23
24 - name: "httpd-vhosts: Incl virtual hosts in {{ apache_conf_path }}/httpd.conf"
25   lineinfile:
26     dest: "{{ apache_conf_path }}/httpd.conf"
27     regexp: "^Include etc/apache{{ apache_version }}/extra/{{ item.ServerName }}.conf"
28     line: "Include etc/apache{{ apache_version }}/extra/{{ item.ServerName }}.conf"
29     insertbefore: "BOF"
30     backup: "{{ apache_backup_conf }}"
31     loop: "{{ apache_vhost }}"
32     notify: reload apache
33
34 # EOF
35 ...

```

Template vhost.j2

Create both http and https servers (1,8). Optionally default (True) redirect permanent http to https (4).

[vhost.j2]

```

1 <VirtualHost *:80>
2   ServerName {{ item.ServerName }}
3   DocumentRoot {{ item.DocumentRoot }}
4   {% if item.redirect|default(True) %} Redirect permanent / https://{{ item.
5   ↪ServerName }}/
6   {% endif %}
7   </VirtualHost>
8
9 <VirtualHost *:443>
10  ServerName {{ item.ServerName }}

```

(continues on next page)

(continued from previous page)

```

10 DocumentRoot {{ item.DocumentRoot }}
11 SSLCertificateFile {{ item.SSLCertificateFile }}
12 SSLCertificateKeyFile {{ item.SSLCertificateKeyFile }}
13 </VirtualHost>

```

See Also

See also:

- Variable *apache_vhost*

2.4.7 Configure confd

- *Synopsis*
- *Annotation*
- *Code*

Synopsis

- Configure virtual hosts

Annotation

Configure virtual hosts in `extra` directory (4). Use `encode_apache` filter. Configure sections in `Includes` directory (8). Take the configuration data from the directories `{{ apache_confdir_vhosts }}` and `{{ apache_confdir_sections }}`. Include the files in `httpd.conf`.

Code

[httpd-confd.yml]

```

1 ---
2
3 - name: "httpd-confd: Configure virtual hosts from {{ apache_confdir_vhosts }}"
4   import_tasks: "httpd-confd-vhosts.yml"
5   tags: apache-httpd-confd-vhosts
6
7 - name: "httpd-confd: Configure includes from {{ apache_confdir_sections }}"
8   import_tasks: "httpd-confd-includes.yml"
9   tags: apache-httpd-confd-includes
10
11 # EOF
12 ...

```


2.4.8 Configure confd-vhosts

- *Synopsis*
- *Annotation*
 - *Include data from conf.d (3-17)*
 - *Create directories for virtual hosts (25-27)*
 - *Configure virtual hosts in extra directory (29-38)*
 - *Include virtual hosts in httpd.conf (40-48)*
 - *Configure confd-vhost-dirs*

Synopsis

- Configure virtual hosts with the filter `encode_apache`.
- Take the YAML configuration files from the directory `{{ apache_conf_dir_vhosts }}` at master and create files with the Apache virtual hosts in `{{ apache_conf_path }}/extra/` at the remote host.
- The created files will be included in `{{ apache_conf_path }}/httpd.conf`.
- The example of the YAML format of the configuration file is available at `vars/conf.d/vhosts-sample/example.com.yml`

Annotation

Include data from conf.d (3-17)

Include tasks from the file `al_include_conf_d_vars_list` (12) in the role `vbotka.ansible_lib` (11). This task takes as parameters the directory with the YAML configuration files (7), and the type of the list (8) and returns the list with the YAML configurations of the virtual hosts stored in the variable `al_include_conf_d_vars_list`. The variable can be printed (17) if debug is turned on `apache_debug: true`. The parameters (7,8) are tested inside the included tasks.

See also:

See details of the included tasks at `al_include_conf_d_vars_list.yml`.

Create directories for virtual hosts (25-27)

Include tasks from `fn/httpd-confd-vhost-dirs.yml`.

See also:

Configure confd-vhost-dirs.

Configure virtual hosts in extra directory (29-38)

Create the Apache configuration files for the virtual hosts with the help of `encode_apache` filter. Store the files in the directory `{{ apache_conf_path }}/extra/`.

See also:

For details see the template `vhost2.j2`.

Include virtual hosts in httpd.conf (40-48)

Include virtual hosts in `httpd.conf`.

[`httpd-confd-vhosts.yml`]

```

1 ---
2
3 - name: "httpd-confd-vhosts: Include variables from {{ apache_conf_dir_vhosts }}"
4   block:
5     - name: "httpd-confd-vhosts: Set variables for al_include_conf_vars_list"
6       set_fact:
7         al_include_conf_dir: "{{ apache_conf_dir_vhosts }}"
8         al_include_conf_vars_list_type: "fname"
9     - name: "httpd-confd-vhosts: Include al_include_conf_vars_list"
10      include_role:
11        name: "vbotka.ansible_lib"
12        tasks_from: "al_include_conf_vars_list"
13    - name: "httpd-confd-vhosts: Debug al_include_conf_vars_list"
14      when: apache_debug|bool
15      debug:
16        var: item
17      loop: "{{ al_include_conf_vars_list }}"
18
19 - name: "httpd-confd-vhosts: Debug list directories for virtual hosts"
20   when: apache_debug|bool
21   debug:
22     msg: "{{ item|json_query('[] .content[] .sections[] .content[] .options[] .DocumentRoot
23 ↪') }}"
24     loop: "{{ al_include_conf_vars_list|json_query('[] .vars') }}"
25
26 - name: "httpd-confd-vhosts: Create directories for virtual hosts"
27   include_tasks: "fn/httpd-confd-vhost-dirs.yml"
28   loop: "{{ al_include_conf_vars_list|json_query('[] .vars') }}"
29
30 - name: "httpd-confd-vhosts: Configure virtual hosts in {{ apache_conf_path }}/extra/"
31   template:
32     src: "vhost2.j2"
33     dest: "{{ apache_conf_path }}/extra/{{ item.fname }}.conf"
34     owner: "{{ apache_data_owner }}"
35     group: "{{ apache_data_group }}"
36     mode: "{{ apache_data_mode }}"
37     backup: "{{ apache_backup_conf }}"
38     loop: "{{ al_include_conf_vars_list }}"
39     notify: reload apache
40
41 - name: "httpd-confd-vhosts: Incl virtual hosts in {{ apache_conf_path }}/httpd.conf"
42   lineinfile:
43     dest: "{{ apache_conf_path }}/httpd.conf"
44     regexp: "^Include etc/apache{{ apache_version }}/extra/{{ item.fname }}.conf"
45     line: "Include etc/apache{{ apache_version }}/extra/{{ item.fname }}.conf"
46     insertbefore: "BOF"
47     backup: "{{ apache_backup_conf }}"

```

(continues on next page)

(continued from previous page)

```

47 loop: "{{ al_include_conf_d_vars_list }}"
48 notify: reload apache
49
50 # EOF
51 ...

```

Configure confd-vhost-dirs

- Create DocumentRoot directories for vhosts.

[fn/httpd-confd-vhost-dirs.yml]

```

1 ---
2 - name: "httpd-confd-vhost-dirs: Create directories for virtual hosts"
3   file:
4     state: "directory"
5     path: "{{ vhost_dir }}"
6     owner: "{{ apache_data_owner }}"
7     group: "{{ apache_data_group }}"
8     mode: "{{ apache_dir_mode }}"
9     loop: "{{ item.json_query('[] .content[] .sections[] .content[] .options[] .DocumentRoot
10 ↪') }}"
11   loop_control:
12     loop_var: vhost_dir
13   ...

```

2.4.9 Configure confd-includes

- *Synopsis*
- *Annotation*
 - *Include data from conf.d (3-17)*
 - *Configure sections in Includes directory (19-28)*
 - *Template section2.j2*

Synopsis

- Configure sections with the filter `encode_apache`.

Annotation

- Take the YAML configuration files from the directory `{{ apache_conf_d_dir_sections }}` at master and create files with the sections in `{{ apache_conf_path }}/Includes/` at the remote host. These created files are included in `{{ apache_conf_path }}/httpd.conf` by default.

```

$ grep Includes /usr/local/etc/apache24/httpd.conf
Include etc/apache24/Includes/*.conf

```

- The example of the YAML format of the configuration file is available at `vars/conf.d/section-sample/usr-local-www-example.com.yml` and the details of the format are described in the filter `encode_apache`.

Include data from conf.d (3-17)

Include tasks from the file `al_include_conf_d_vars_list` (12) in the role `vbotka.ansible_lib` (11). This task takes as parameters the directory with the YAML configuration files (7), and the type of the list (8) and returns the list with the YAML configurations of the sections stored in the variable `al_include_conf_d_vars_list`. The variable can be printed (17) if debug is turned on `apache_debug: true`. The parameters (7,8) are tested inside the included tasks.

See also:

See details of the included tasks at `al_include_conf_d_vars_list.yml`.

Configure sections in Includes directory (19-28)

Create the Apache configuration files for the sections with the help of `encode_apache` filter. Store the files in the directory `{{ apache_conf_path }}/Includes/` (22).

[`httpd-confd-includes.yml`]

```

1  ---
2
3  - name: "httpd-confd-includes: Include sections from {{ apache_conf_d_dir_sections }}"
4    block:
5      - name: "httpd-confd-includes: Set variables for al_include_conf_d_vars_list"
6        set_fact:
7          al_include_conf_d_dir: "{{ apache_conf_d_dir_sections }}"
8          al_include_conf_d_vars_list_type: "fname"
9      - name: "httpd-confd-includes: Include al_include_conf_d_vars_list"
10       include_role:
11         name: vbotka.ansible_lib
12         tasks_from: al_include_conf_d_vars_list
13      - name: "httpd-confd-includes: Debug al_include_conf_d_vars_list"
14        when: apache_debug|bool
15        debug:
16          var: item
17        loop: "{{ al_include_conf_d_vars_list }}"
18
19 - name: "httpd-confd-includes: Configure sections in {{ apache_conf_path }}/Includes/"
20   template:
21     src: "section2.j2"
22     dest: "{{ apache_conf_path }}/Includes/{{ item.fname }}.conf"
23     owner: "{{ apache_data_owner }}"
24     group: "{{ apache_data_group }}"
25     mode: "{{ apache_data_mode }}"
26     backup: "{{ apache_backup_conf }}"
27     loop: "{{ al_include_conf_d_vars_list }}"
28     notify: reload apache
29
30 # EOF
31 ...

```

Template section2.j2

[section2.j2]

```

1  {% for section in item.vars %}
2  {{ section | encode_apache }}
3  {% endfor %}

```

2.5 Service

[service.yml]

```

1  ---
2
3  - name: "service: Config FreeBSD"
4    import_tasks: "rcconf.yml"
5    when: ansible_os_family == "FreeBSD"
6
7  # EOF
8  ...

```

[rcconf.yml]

```

1  ---
2
3  - name: "rc_conf: Enable and Start Apache"
4    when: apache_enable|bool
5    lineinfile:
6      dest: "/etc/rc.conf"
7      regexp: "^\\s*{{ apache_service }}_enable\\s*="
8      line: "{{ apache_service }}_enable=\"YES\""
9      backup: "{{ apache_backup_conf }}"
10     notify: enable and start apache
11
12  - name: "rc_conf: Disable and Stop Apache"
13    when: not apache_enable|bool
14    lineinfile:
15      dest: "/etc/rc.conf"
16      regexp: "^\\s*{{ apache_service }}_enable\\s*="
17      line: "{{ apache_service }}_enable=\"NO\""
18      backup: "{{ apache_backup_conf }}"
19     notify: disable and stop apache
20
21  # EOF
22  ...

```

2.6 Handlers

[main.yml]

```

1  ---
2  # handlers for vbotka.apache

```

(continues on next page)

(continued from previous page)

```
3
4 - name: "enable and start apache"
5   service:
6     name: "{{ apache_service }}"
7     state: "started"
8     enabled: true
9
10 - name: "disable and stop apache"
11   service:
12     name: "{{ apache_service }}"
13     state: "stopped"
14     enabled: false
15
16 - name: "restart apache"
17   when: apache_enable|bool
18   service:
19     name: "{{ apache_service }}"
20     state: "restarted"
21
22 - name: "reload apache"
23   when: apache_enable|bool
24   service:
25     name: "{{ apache_service }}"
26     state: "reloaded"
27
28 - name: "graceful apache"
29   when: apache_enable|bool
30   command: "/usr/local/etc/rc.d/{{ apache_service }} graceful"
31
32 - name: "gracefulstop apache"
33   when: apache_enable|bool
34   command: "/usr/local/etc/rc.d/{{ apache_service }} gracefulstop"
35
36 # EOF
37 ...
```

[host_vars/srv.example.com/apache.yml]

```
1 ---
2 # vars for apache
3
4 apache_enable: true
5 apache_version: "24"
6 apache_php_lib: "7"
7
8 al_debug: true
9 apache_debug: true
10 apache_backup_conf: true
11
12 apache_ssl: true
13 apache_php: true
14 apache_vhosts: true
15 apache_SSLEngine: "on"
16
17 apache_packages:
18   - "{{ apache_package }}"
19   - "{{ apache_php_package }}"
20
21 apache_ServerName: "srv.example.com"
22 apache_ServerAdmin: "admin@example.com"
23 apache_SSLCertificateFile: "/usr/local/etc/ssl/certs/srv.example.com.crt"
24 apache_SSLCertificateKeyFile: "/usr/local/etc/ssl/private/srv.example.com.key"
25
26 apache_confdir_vhosts: "{{ playbook_dir }}/servers/apache.d/vhosts"
27 apache_confdir_sections: "{{ playbook_dir }}/servers/apache.d/sections"
28
29 apache_httpd_conf_modules:
30   - {module: "socache_shmcb_module", mod: "mod_socache_shmcb.so"}
31   - {module: "ssl_module", mod: "mod_ssl.so"}
32   - {module: "php{{ apache_php_lib }}_module", mod: "libphp{{ apache_php_lib }}.so"}
```

(continues on next page)

(continued from previous page)

```
33 - {module: "cgi_module", mod: "mod_cgi.so", present: true}
34 - {module: "cgid_module", mod: "mod_cgid.so", present: true}
35
36 apache_directory_blocks: {}
37 apache_vhost: {}
38 apache_alias:
39 - "ScriptAlias /nagios/cgi-bin/ /usr/local/www/nagios/cgi-bin/"
40 - "Alias /nagios/ /usr/local/www/nagios/"
41
42 # EOF
43 ...
```


4.1 Requirements

- `vbotka.ansible_lib`
- `jtyr.config_encoder_filters`

4.2 Default variables

- `defaults/main.yml`
- `vars/defaults/FreeBSD.yml`

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`