Roles

- BookStack
- CorwdSec
- n8n

BookStack

Status	Production
Supported distro	CentOS 7/8

This roles deploys the BookStack application

Backend server

Sample backend server

```
httpd_ansible_vhosts:
  - name: docs.example.org
    document_root: /opt/bookstack_1/app/public

bookstack_public_url: https://docs.example.org
bookstack_web_alias: False
bookstack_settings:
  AUTH_METHOD: saml2
  SAML2_NAME: EXAMPLE
  SAML2_DISPLAY_NAME_ATTRIBUTES: cn
  SAML2_EXTERNAL_ID_ATTRIBUTE: principal
  SAML2_IDP_ENTITYID: https://sso.example.org/saml/metadata
  SAML2_AUTOLOAD_METADATA: 'false'
  SAML2_IDP_SSO: https://sso.example/saml/singleSignOn
  SAML2_IDP_x509: MIICpjCCAY6gAXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  SAML2_USER_TO_GROUPS: 'true'
  SAML2_GROUP_ATTRIBUTE: groups
  SAML2_REMOVE_FROM_GROUPS: 'true'
  DRAWIO: https://draw.example.org/?embed=1&proto=json&spin=1
```

Reverse proxy

Sample reverse proxy config, using the nginx role

```
nginx_vhosts:

  # BookStack
  - name: docs.fws.fr
    allowed_methods: [GET,HEAD,POST,OPTIONS,PUT,DELETE]
    csp: >-
      default-src 'self' 'unsafe-inline' blob:;
      style-src-elem 'self' 'unsafe-inline' data:;
      img-src 'self' data: blob: https://stats.example.org;
      script-src 'self' 'unsafe-inline' 'unsafe-eval' https://stats.example.org blob:;
      font-src 'self' data:;
      frame-src https://sso.example.org https://draw.example.org
    src_ip:
      - "{{ trusted_ip }}"
```

CorwdSec

Status	Production
Supported distro	CentOS 7/8

This role deploys the crowdsec agent.

CrowdSec agent

Here's a sample config for the agent. Running with local and central API enabled, using a MySQL bakend

```
cs_db_engine: mysql
cs_capi_enabled: True
cs_lapi_enabled: True
cs_lapi_src_ip:
  - 10.29.1.14 # Only the rev proxy can reach the API
cs_use_x_forwarded_for: True
cs_acquis:
  - filenames:
      - /run/g2cs/logs/syslog.log
    labels:
      type: syslog
  - filenames:
      - /run/g2cs/logs/nginx/access.log
      - /run/g2cs/logs/nginx/error.log
    labels:
      type: nginx
  - filenames:
      - /run/g2cs/logs/httpd/access.log
      - /run/g2cs/logs/httpd/error.log
    labels:
      type: apache2
  - filenames:
      - /run/g2cs/logs/zimbra/mailbox.log
```

```
    labels:
      type: zimbra

cs_parsers:
  - crowdsecurity/syslog-logs
  - crowdsecurity/geoip-enrich
  - crowdsecurity/dateparse-enrich
  - crowdsecurity/whitelists
  - crowdsecurity/sshd-logs
  - crowdsecurity/iptables-logs
  - crowdsecurity/nginx-logs
  - crowdsecurity/apache2-logs
  - crowdsecurity/http-logs
  - crowdsecurity/postfix-logs
  - crowdsecurity/postscreen-logs
  - firewallservices/zimbra-logs
  - firewallservices/lemonldap-ng
cs_scenarios:
  - crowdsecurity/ban-defcon-drop_range
  - crowdsecurity/ssh-bf
  - crowdsecurity/http-bf-wordpress_bf
  - crowdsecurity/http-generic-bf
  - crowdsecurity/http-sensitive-files
  - crowdsecurity/http-sqli-probing
  - crowdsecurity/http-xss-probing
  - crowdsecurity/http-backdoors-attempts
  - crowdsecurity/http-path-traversal-probing
  - crowdsecurity/iptables-scan-multi_ports
  - crowdsecurity/postfix-spam
  - firewallservices/zimbra-bf
  - firewallservices/lemonldap-ng-bf
```

g2cs

You can use the g2cs companion role. This small daemon can receive a syslog stream from graylog (in CEF format) and write files for crowdsec to consume

```
cs_user: g2cs
```

```
g2cs_port: 514
g2cs_src_ip:
  - 10.29.1.20 # Graylog FWS
```

CrowdSec Firewall Bouncer

You can deploy firewall bouncers with the crowdsec_firewall_bouncer role

```
# Use crowdsec local API
cs_lapi_url: https://cs.example.org/
cs_lapi_server: crowdsec.example.org
```

Reverse proxy

You can use a reverse proxy to expose your local API to your other machines. Here's a sample on a machine using the nginx role to provide the reverse proxy functionnality

```
nginx_vhosts:
  # Crowdsec Local API
  - name: cs.example.org
    proxy:
      backend: http://crowdsec.example.org:8080
    auth: False
    src_ip:
      - "{{ trusted_ip }}"
      - 10.29.0.0/16
      - 10.30.0.0/16
      - 10.99.0.0/16
      - 192.168.7.0/24
```

n8n

Status	Production
Supported distro	CentOS 8

This role deploys the n8n workflow manager.

Here's a sample configuration

```
n8n_src_ip:
  - 10.29.1.14 # Access only for the rev proxy
n8n_public_url: https://workflow.example.org/
```

And a sample reverse proxy configurtion, using the nginx role

```
nginx_vhosts:
  # n8n
  - name: workflow.example.org
    proxy:
      backend: http://n8n.fws.fr:8021
    allowed_methods: [GET,HEAD,POST,OPTIONS,PUT,DELETE,PATCH]
```