Installation et configuration de Wazuh

Téléchargement et installation du paquet

```
curl -sO https://packages.wazuh.com/4.3/wazuh-install.sh && sudo bash
./wazuh-install.sh -a
```

Accès interface web

https://@IPSrv|FQDN

Déploiement d'agent

Windows

```
Invoke-WebRequest -Uri
https://packages.wazuh.com/4.x/windows/wazuh-agent-4.3.10-1.msi -OutFile
${env:tmp}\wazuh-agent-4.3.10.msi; msiexec.exe /i
${env:tmp}\wazuh-agent-4.3.10.msi /q WAZUH_MANAGER='<@IPServeur>'
WAZUH_REGISTRATION_SERVER='<@IPServeur>' WAZUH_PROTOCOL='TCP'
WAZUH_AGENT_GROUP='default'
```

Debian/Ubuntu

```
curl -so wazuh-agent-4.3.10.deb
https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4
.3.10-1_amd64.deb && sudo WAZUH_MANAGER='<@IPServeur>'
WAZUH_PROTOCOL='TCP' WAZUH_AGENT_GROUP='default' dpkg -i
./wazuh-agent-4.3.10.deb
```

Théo PERDIGON

MacOS

```
curl -so wazuh-agent-4.3.10.pkg
https://packages.wazuh.com/4.x/macos/wazuh-agent-4.3.10-1.pkg && sudo
launchctl setenv WAZUH_MANAGER '<@IPServeur>' WAZUH_PROTOCOL 'TCP'
WAZUH_AGENT_GROUP 'default' && sudo installer -pkg
./wazuh-agent-4.3.10.pkg -target /
```

Activation des notifications par e-mail

1 - Installation des paquets nécessaires

```
apt-get install postfix mailutils libsasl2-2 ca-certificates
libsasl2-modules
```

2 - Configuration de Postfix

Modification du fichier `/etc/postfix/main.cf` :

```
# Configuration manuelle

relayhost = <ServeurSMTP:Port>
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl/sasl_passwd
smtp_sasl_security_options = noanonymous
smtp_generic_maps = hash:/etc/postfix/generic
smtp_tls_wrappermode = yes
smtp_tls_security_level = encrypt
```

Création du fichier `/etc/postfix/sasl/sasl_passwd` :

```
<ServeurSMTP> <AdresseMailEnvoi>:<MotDePasse>
```

Hachage du fichier :

```
postmap hash:/etc/postfix/sasl/sasl_passwd
```

Création du fichier `/etc/postfix/generic` :

```
root@<FQDN> <AdresseMailEnvoi>
```

Création du hash :

```
postmap hash:/etc/postfix/generic
```

Modification du fichier `/etc/aliases` :

```
root  <AdresseMailEnvoi>
root@<FQDN> <AdresseMailEnvoi>
```

Redémarrage de Postfix :

```
systemctl restart postfix
```

3 - Configuration de Wazuh

Modification du fichier `/var/ossec/etc/ossec.conf` :

```
<global>
    <email_notification>yes</email_notification>
    <smtp_server>localhost</smtp_server>
    <email_from>AdresseEmailEnvoi</email_from>
    <email_to>AdresseMailRéception</email_to>
    <email_maxperhour>12</email_maxperhour>
</global>

...

<alerts>
    <log_alert_level>3</log_alert_level>
    <email_alert_level>12</email_alert_level>
</alerts>
```

Redémarrage de Wazuh Manager :

```
systemctl restart wazuh-manager.service
```

Activation de l'authentification via certificat

1 - Serveur Wazuh

Modification du fichier `/var/ossec/etc/ossec.conf` :

```
<auth>
   ...
   <ssl_agent_ca>CheminVersLaCA</ssl_agent_ca>
   <ssl_verify_host>no</ssl_verify_host>
   <ssl_manager_cert>CheminVersLeCert</ssl_manager_cert>
   <ssl_manager_key>CheminVersLaClé</ssl_manager_key>
   ...
 <auth>
```

Redémarrage du service Wazuh-manager

```
systemctl restart wazuh-manager.service
```

2 - Agent Wazuh

Modification du fichier `/var/ossec/etc/ossec.conf` :

```
  <enrollment>
     ...
     <agent_certificate_path>CheminVersLeCert</agent_certificate_path>
     <agent_key_path>CheminVersLaClé</agent_key_path>
     ...
   </enrollment>
```

Redémarrage de l'agent

```
systemctl restart wazuh-agent.service
```

Surveillance d'un service en particulier

Modification du fichier de configuration `/var/ossec/etc/ossec/conf` :

```
<localfile>
   <log_format>syslog</log_format>
   <location>/var/log/<Service.log></location>
 </localfile>
```

Théo PERDIGON

Activation de l'authentification via LDAPs

Modification du fichier `/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/config.yml` :

```yaml
---
_meta:
  type: "config"
  config_version: 2
config:
  dynamic:
    http:
      anonymous_auth_enabled: false
      xff:
        enabled: false
        internalProxies: "192\\.168\\.0\\.10|192\\.168\\.0\\.11"
    authc:
      basic_internal_auth_domain:
        description: "Authenticate via HTTP Basic against internal users database"
        http_enabled: true
        transport_enabled: true
        order: 0
        http_authenticator:
          type: "basic"
          challenge: false
        authentication_backend:
          type: "intern"
      ldap:
        description: "Authenticate via LDAP or Active Directory"
        http_enabled: true
        transport_enabled: true
        order: 1
        http_authenticator:
          type: "basic"
          challenge: false
        authentication_backend:
          type: "ldap"
          config:
            enable_ssl: true
            pemtrustedcas_filepath: <CheminVersCertificatCA>
            enable_start_tls: false
            enable_ssl_client_auth: false
            verify_hostnames: true
```

```
        hosts:
        - "<ServerLDAPs>:636"
        bind_dn: "cn=reader,ou=people,dc=domain,dc=lan"
        password: "<MotDePasse>"
        userbase: "ou=people,dc=domain,dc=lan"
        usersearch: "(uid={0})"
        username_attribute: "cn"
  authz:
    roles_from_myldap:
      description: "Authorize via LDAP or Active Directory"
      http_enabled: true
      transport_enabled: true
      authorization_backend:
        type: "ldap"
        config:
          enable_ssl: true
          pemtrustedcas_filepath: <CheminVersCertificatCA>
          enable_start_tls: false
          enable_ssl_client_auth: false
          verify_hostnames: true
          hosts:
          - "<ServeurLDAPs>:636"
          bind_dn: "cn=reader,ou=people,dc=domain,dc=lan"
          password: "<MotDePasse>"
          rolebase: "ou=groups,dc=domain,dc=lan"
          rolesearch: "(member={0})"
          userroleattribute: null
          userrolename: "member"
          rolename: "cn"
          resolve_nested_roles: false
          skip_users:
          - "admin"
          - "kibanaserver"
          userbase: "ou=people,dc=domain,dc=lan"
          usersearch: "(uid={0})"
          username_attribute: "cn"
```

Modification du fichier `/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/roles_mapping.yml` :

```
all_access:
  reserved: false
  hidden: false
  backend_roles:
  - "admin"
  - "<NomGroupeLDAP>"
  hosts: []
  users: []
  and_backend_roles: []
  description: "Maps admin to all_access"
```

Envoi des fichiers dans la configuration d'OpenSearch :

```
export JAVA_HOME=/usr/share/wazuh-indexer/jdk/ && bash
/usr/share/wazuh-indexer/plugins/opensearch-security/tools/securityadmin
.sh -f
/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/conf
ig.yml -icl -key /etc/wazuh-indexer/certs/admin-key.pem -cert
/etc/wazuh-indexer/certs/admin.pem -cacert
/etc/wazuh-indexer/certs/root-ca.pem -h 127.0.0.1 -nhnv

export JAVA_HOME=/usr/share/wazuh-indexer/jdk/ && bash
/usr/share/wazuh-indexer/plugins/opensearch-security/tools/securityadmin
.sh -f
/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/role
s_mapping.yml -icl -key /etc/wazuh-indexer/certs/admin-key.pem -cert
/etc/wazuh-indexer/certs/admin.pem -cacert
/etc/wazuh-indexer/certs/root-ca.pem -h 127.0.0.1 -nhnv
```

Redémarrage de Wazuh-indexer et Wazuh-dashboard

```
systemctl restart wazuh-indexer wazuh-dashboard
```

Activation de l'authentification via SAML

Modification du fichier `/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/config.yml` :

```yaml
---
_meta:
  type: "config"
  config_version: 2
config:
  dynamic:
    http:
      anonymous_auth_enabled: false
      xff:
        enabled: false
        internalProxies: "192\\.168\\.0\\.10|192\\.168\\.0\\.11"
    authc:
      basic_internal_auth_domain:
        description: "Authenticate via HTTP Basic against internal users database"
        http_enabled: true
        transport_enabled: true
        order: 0
        http_authenticator:
          type: "basic"
          challenge: false
        authentication_backend:
          type: "intern"
      saml_auth_domain:
        http_enabled: true
        transport_enabled: false
        order: 1
        http_authenticator:
          type: "saml"
          challenge: true
          config:
            idp:
              entity_id: <IDP>
              metadata_file: <CheminVersLeFichier>
            sp:
              entity_id: <SP>
            kibana_url: <UrlDashboard>
            exchange_key: <Chaine32Caractères>
            roles_key: <Role/GroupeLDAP>
```

```
        authentication_backend:
           type: noop
...
```

Modification du fichier `/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/roles_mapping.yml` :

```
all_access:
  reserved: false
  hidden: false
  backend_roles:
  - "admin"
  - "<RoleAMapper>"
  hosts: []
  users: []
  and_backend_roles: []
  description: "Maps admin to all_access"
```

Envoi des fichiers dans la configuration d'OpenSearch :

```
export JAVA_HOME=/usr/share/wazuh-indexer/jdk/ && bash
/usr/share/wazuh-indexer/plugins/opensearch-security/tools/securityadmin
.sh -f
/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/conf
ig.yml -icl -key /etc/wazuh-indexer/certs/admin-key.pem -cert
/etc/wazuh-indexer/certs/admin.pem -cacert
/etc/wazuh-indexer/certs/root-ca.pem -h 127.0.0.1 -nhnv

export JAVA_HOME=/usr/share/wazuh-indexer/jdk/ && bash
/usr/share/wazuh-indexer/plugins/opensearch-security/tools/securityadmin
.sh -f
/usr/share/wazuh-indexer/plugins/opensearch-security/securityconfig/role
s_mapping.yml -icl -key /etc/wazuh-indexer/certs/admin-key.pem -cert
/etc/wazuh-indexer/certs/admin.pem -cacert
/etc/wazuh-indexer/certs/root-ca.pem -h 127.0.0.1 -nhnv
```

Modification du fichier `/etc/wazuh-dashboard/opensearch_dashboards.yml` :

```
opensearch_security.auth.type: "saml"
server.xsrf.whitelist:
[/_plugins/_security/saml/acs,/_opendistro/_security/saml/acs,/_plugins/
_security/saml/acs/idpinitiated,/_opendistro/_security/saml/acs/idpiniti
ated,/_plugins/_security/saml/logout,/_opendistro/_security/saml/logout]
```

Redémarrage de Wazuh-indexer et Wazuh-dashboard

```
systemctl restart wazuh-indexer wazuh-dashboard
```