Ansible

Cybersécurité et Automatisation avec Ansible

Introduction

Dans un monde où les cybermenaces évoluent constamment, la capacité à déployer rapidement des mesures de sécurité, à gérer la configuration de systèmes et à réagir efficacement aux incidents est cruciale. L'automatisation est devenue un pilier fondamental de la cybersécurité moderne, permettant aux équipes de sécurité de gagner en efficacité, de réduire les erreurs humaines et d'assurer une meilleure conformité.

Vous faites partie d'une équipe d'administrateurs système et de sécurité au sein de Starfleet, une entreprise qui gère une infrastructure Linux variée. Votre mission est de démontrer la puissance d'Ansible, un outil d'automatisation sans agent, pour renforcer la posture de sécurité de leurs serveurs, gérer les configurations et simuler des réponses à incident.

Job 1 - Prise en Main et Premières Automatisations

1. Installation d'Ansible : Installez Ansible sur votre serveur de contrôle.

2. Configuration SSH : Générez une paire de clés SSH sur le contrôleur et distribuez la clé publique sur toutes vos machines cibles pour une connexion sans mot de passe.

3. Création de l'Inventaire : Créez un fichier inventory.ini regroupant vos machines cibles par catégories (ex: [webservers], [dbservers], [all_servers]).

4. Tests Initiaux : Vérifiez la connectivité avec vos machines via des commandes ansible all -m ping et exécutez de simples commandes à distance pour confirmer le bon fonctionnement.

5. Premiers Playbooks :

- Créez un playbook pour vérifier l'état des services sur toutes les cibles.
- Créez un playbook pour copier un fichier simple depuis votre contrôleur vers les machines cibles.
- Développez un playbook pour mettre à jour tous les paquets du système sur vos machines cibles, en tenant compte des différentes distributions Linux (apt pour Debian/Ubuntu, yum pour CentOS/Rocky Linux).

Job 2 - Hardening des Systèmes

L'objectif est d'appliquer des mesures de sécurité essentielles sur vos serveurs.

1. Gestion Sécurisée des Utilisateurs et Groupes :

- Créez un playbook pour supprimer des utilisateurs par défaut considérés comme inutiles ou à risque sur les systèmes (recherchez quels utilisateurs système pourraient être ciblés).

◦ Créez un playbook pour créer un nouvel utilisateur administrateur dédié (ansible_admin) avec un mot de passe haché et ajoutez-le au groupe sudo ou wheel.

2. Durcissement de la Configuration SSH :

◦ Développez un playbook qui modifie le fichier /etc/ssh/sshd_config sur toutes les machines cibles pour :

▪ Désactiver la connexion de l'utilisateur root.

▪ Désactiver l'authentification par mot de passe.

▪ Changer le port SSH par défaut (par exemple, vers le port 2002).

▪ S'assurer que X11Forwarding est désactivé.

◦ Redémarrez le service SSH après ces modifications et testez votre connexion avec les nouvelles règles.

3. Configuration du Pare-feu :

◦ Créez un playbook pour installer et configurer le service de pare-feu approprié (ufw).

◦ Activez le pare-feu et définissez une politique par défaut pour bloquer le trafic entrant.

◦ Ajoutez des règles pour autoriser les connexions sur le nouveau port SSH (2002), ainsi que les ports HTTP (80) et HTTPS (443).

4. Désactivation des services inutiles :

◦ Identifiez les services système par défaut qui ne sont pas essentiels pour une machine serveur et créez un playbook pour les désactiver et les masquer (par exemple, cups, avahi-daemon). Soyez prudent lors de cette étape.

Job 3 - Surveillance, Logs et Conformité

Cette mission se concentre sur l'intégration d'outils de surveillance et l'audit de sécurité.

1. Déploiement de Filebeat pour la Gestion des Logs :

◦ Créez un playbook pour installer et configurer Filebeat sur toutes les machines cibles.

◦ Le playbook devra gérer l'installation du paquet Filebeat et la configuration de son fichier filebeat.yml (utilisez un template Jinja2 pour cela). Configurez-le pour activer les modules system et auditd (si applicable) et pour simuler l'envoi des logs vers une destination fictive.

◦ Démarrez et activez le service Filebeat.

2. Audit de Conformité Basique :

◦ Développez un playbook pour effectuer des vérifications de conformité de base, notamment :

▪ Vérifier l'existence et les permissions de fichiers système critiques (/etc/passwd, /etc/shadow, /etc/sudoers, /etc/ssh/sshd_config).

▪ Identifier si des utilisateurs ont un mot de passe vide.

▪ Rechercher des répertoires avec des permissions trop permissives (ex: 777).

3. Durcissement du Noyau et des Politiques de Mot de Passe :

◦ Créez un playbook pour configurer des paramètres de durcissement du noyau via /etc/sysctl.conf (ex: net.ipv4.tcp_syncookies = 1, net.ipv4.ip_forward = 0).

◦ Modifiez les politiques de mot de passe via PAM (/etc/pam.d/common-password ou /etc/login.defs) pour forcer une longueur minimale, la complexité des caractères, et l'historique des mots de passe.

4. Introduction aux Rôles Ansible :

◦ Commencez à structurer votre travail en rôle Ansible. Créez au moins un rôle (ex: hardening_base) et déplacez les tâches pertinentes dans ce rôle pour une meilleure organisation et réutilisabilité.

Job 4 - Détection, Réponse à Incident et Cas d'Usage Avancés

Explorez des cas d'utilisation plus complexes d'Ansible pour la détection et la réponse.

1. Déploiement d'un IDS/IPS Simple (Snort ou Suricata) :

◦ Créez un playbook pour installer un système de détection d'intrusion (IDS) comme Snort ou Suricata sur une ou plusieurs machines cibles. Configurez-le de manière basique pour qu'il surveille une interface réseau.

2. Scénario de Réponse à Incident (simulé) :

◦ Préparation : Sur une de vos machines cibles, créez manuellement un "fichier suspect" (ex: /tmp/malicious_script.sh).

◦ Créez un playbook qui simule une réponse à incident :

▪ Détecte la présence du fichier suspect.

▪ Si détecté, effectue les actions suivantes :

• Arrête un service potentiellement compromis (ex: un service web).

• Déplace le fichier suspect vers un répertoire de quarantaine.

• Collecter des informations de diagnostic (logs récents, liste des processus en cours).

• Envoie une notification (via un message dans un fichier de log sur le contrôleur si l'envoi d'e-mail n'est pas possible).

3. Utilisation d'Ansible Vault :

◦ Démontrez l'utilisation d'Ansible Vault pour chiffrer des informations sensibles (ex: un mot de passe, une clé API fictive) au sein de vos playbooks.

4. Gestion des Faits (Facts) :

◦ Utilisez le module setup pour collecter des informations détaillées sur les systèmes cibles et écrivez une tâche simple pour afficher des faits spécifiques (ex: la version du système d'exploitation, la quantité de RAM).

Job 5 - Scénario, Optimisation et Bilan

Intégrez toutes les compétences acquises dans un déploiement complet et présentez votre travail.

1. Scénario de Déploiement Sécurisé d'une Application Web Simple :

◦ Cahier des Charges : Déployez une application web simple (une page HTML statique) sur un de vos serveurs en utilisant Nginx.

◦ Votre playbook devra :

▪ Appliquer toutes les mesures de hardening développées précédemment sur ce serveur.

▪ Installer Nginx.

- Déployer la page HTML statique.
- Configurer Nginx pour servir cette page et utiliser HTTPS avec un certificat auto-signé.
- S'assurer que les logs de Nginx sont collectés par Filebeat.
- Redémarrer Nginx.

◦ Testez l'accès à votre page web via HTTPS.

2. Optimisation et Structure des Playbooks :

◦ Passez en revue tous vos playbooks et rôles.

◦ Utilisez des blocs pour regrouper les tâches et des handlers pour les services.

◦ Rendez vos playbooks plus génériques en utilisant des variables et des templates Jinja2.

◦ Assurez-vous que vos rôles sont bien appliqués et que votre projet Ansible est bien organisé.

3. Rapport et Présentation Finale :

◦ Rédigez un rapport synthétique qui inclut :

- Une introduction sur l'importance de l'automatisation en cybersécurité.
- La description de votre environnement de TP.
- Un plan de votre architecture Ansible (inventaire, rôles, playbooks principaux).
- Une explication des mesures de sécurité mises en place et leur justification.
- Les défis rencontrés et les solutions apportées.
- Vos conclusions sur l'efficacité d'Ansible pour la cybersécurité.

◦ Préparez une courte présentation pour montrer votre travail et les concepts appris.

Compétences visées

- Administration Système
- Administration et sécurité réseaux
- Déploiement & mise en production

Rendu

Le projet est à rendre sur votre github :

https://github.com/prenom-nom/bidule

Base de connaissances

- Ansible playbooks
- Deploy Filebeat using Ansible
-