Protocole SRP avec ECDHE

Explication détaillée

1 Introduction

Le **protocole SRP (Secure Remote Password)** est un protocole d'authentification sécurisée conçu pour permettre à deux parties de vérifier mutuellement leur identité de manière sécurisée tout en communiquant sur un réseau potentiellement non sécurisé. Lorsqu'on modifie SRP pour utiliser **ECDHE (Elliptic Curve Diffie-Hellman Ephemeral)**, on bénéficie des avantages des courbes elliptiques en termes de sécurité et de performance.

2 Initialisation

Lors de l'inscription ou de la création d'un compte, l'utilisateur et le serveur effectuent les étapes suivantes :

- **Utilisateur** :
  - Choisit un mot de passe P.
  - Calcule un sel aléatoire s.
  - Calcule un vérificateur v : $v = g^{H(s,P)} \mod N$, où g est une base (générateur), N est un grand nombre premier, et H est une fonction de hachage cryptographique.
- **Serveur** :
  - Stocke le sel s et le vérificateur v dans la base de données.

3 Phase d'authentification

3.1 Initialisation de l'authentification

- **Utilisateur** :
  - Génère une clé privée éphémère a et une clé publique $A = g^a \mod N$.
- **Serveur** :
  - Génère une clé privée éphémère b et une clé publique $B = kv + g^b \mod N$.

- **Échange** :
  - L'utilisateur envoie A au serveur.
  - Le serveur envoie B et s à l'utilisateur.

3.2 Utilisation de l'ECDHE pour la clé de session

Pour intégrer ECDHE, chaque partie génère une paire de clés supplémentaire pour l'échange éphémère :

- **Utilisateur** :
  - Génère une paire de clés ECDHE (d_U, Q_U), où d_U est la clé privée et $Q_U = d_U \cdot G$ est la clé publique, avec G étant le générateur de la courbe elliptique.
  - Envoie Q_U au serveur.
- **Serveur** :
  - Génère une paire de clés ECDHE (d_S, Q_S), où d_S est la clé privée et $Q_S = d_S \cdot G$ est la clé publique.
  - Envoie Q_S à l'utilisateur.

3.3 Calcul des clés de session avec ECDHE

- **Utilisateur** :
  - Calcule la clé partagée $Z_U = d_U \cdot Q_S$.
  - Calcule $u = H(A, B)$.
  - Calcule la clé de session $S_U = (B - kv)^{(a+ux)} \cdot Z_U \mod N$, où $x = H(s, P)$.
- **Serveur** :
  - Calcule la clé partagée $Z_S = d_S \cdot Q_U$.
  - Calcule $u = H(A, B)$.
  - Calcule la clé de session $S_S = (Av^u)^b \cdot Z_S \mod N$.

4 Génération de la clé de session

- **Utilisateur et serveur** :
  - Dérivent la clé de session K en appliquant une fonction de hachage à la clé de session calculée S.

5 Vérification mutuelle

Pour s'assurer que les deux parties possèdent la même clé de session K, elles échangent des preuves de connaissance.

- **Utilisateur** :
  - Calcule $M_U = H(H(N) \oplus H(g) \| H(I) \| s \| A \| B \| K)$, où I est l'identité de l'utilisateur.
  - Envoie M_U au serveur.
- **Serveur** :
  - Calcule M_U de manière indépendante et le compare avec celui reçu de l'utilisateur.
  - Si les valeurs correspondent, le serveur envoie $M_S = H(A \| M_U \| K)$ à l'utilisateur.
- **Utilisateur** :
  - Calcule M_S de manière indépendante et le compare avec celui reçu du serveur.

6 Avantages de l'utilisation de l'ECDHE dans SRP

- **Secret parfait (forward secrecy)** : Les clés éphémères utilisées dans ECDHE assurent que même si les clés privées de longue durée sont compromises, les sessions passées restent sécurisées.
- **Efficacité cryptographique** : Les courbes elliptiques offrent une sécurité équivalente avec des tailles de clé plus petites, ce qui réduit le temps de calcul et l'utilisation de la bande passante.
- **Sécurité accrue** : L'utilisation d'ECDHE ajoute une couche supplémentaire de sécurité, compliquant davantage les tentatives d'attaque.

Ainsi, le protocole SRP avec ECDHE combine les avantages de l'authentification par mot de passe sécurisée de SRP avec les bénéfices de la cryptographie moderne des courbes elliptiques, offrant une solution robuste et efficace pour les connexions sécurisées.