Corosync - Présentation

Présentation du moteur de cluster Corosync

Description

Le projet corosync est un projet pour implémenter un outil de développement de haute disponibilité haute performance et faible charge. Le focus majeur de la haute disponibilité dans le passé a été de masquer les erreurs hardware. Les pannes dans d'autres composants du système restaient non-résolus jusqu'à corosync. Corosync est conçu pour les applications pour répliquer leur état jusqu'à 16 processeurs. Les processeurs contiennent tous un réplica de l'état de l'application.

Le projet corosync fournis une API de message de groupe appelé CPG, qui implément un modèle de messagerie de groupe fermé présentant des garantie de synchronisation virtuel garantie.

Pour gérer les conditions où les processus exécutant l'échange CPG plante, on fournis le Simple Availability Manager (SAM) pour permet de redémarrer l'application.

Démarrage rapide

Dans le répertoire conf dans les sources se trouvent de nombreux fichiers qui doivent être copiés dans le répertoire **/etc/corosync**. corosync devrait fonctionner avec la configuration par défaut. Corosync utilise des techniques cryptographiques pour s'assurer de l'authenticité et de la protection des messages. pour que corosync soit sécurisé, une clé privée doit être générée et partagée par tous les processeurs.

Variables d'environnement

COROSYNC_MAIN_CONFIG_FILE Spécifie le fqdn du fichier de configuration. défaut : /etc/corosync/corosync.conf

COROSYNC_TOTEM_AUTHKEY_FILE fqdn de la clé partagée utilisée pour authentifier et chiffrer les données utilisée dans le protocole Totem. Défaut : /etc/corosync/authkey

Sécurité

Corosync chiffre tous les messages envoyés sur le réseau en utilisant le chiffrement SOBBER-128. Corosync utilise HMAC et SHA1 pour authentifier tous les messages. Corosync utilise NSS comme générateur de nombres pseudo-aléatoire. La libraire EVS utilise /dev/random.

Si les messages de membre peuvent être capturés par des intrus, il est possible d'exécuter une attaque DOS dans le cluster. Dans ce scénario, le cluster est déjà compromis et une attaque DOS est le dernier des soucis de l'administrateur.

La sécurité dans corosync n'offre pas de solution parfaite puisque les clés sont réutilisée. Il peut être possible pour un attaquant de capturer des packets et de déterminer la clé partagée. Dans ce scénario, le cluster est compromis. Pour des raisons de sécurité, corosync ne devrait jamais avoir le setuid ou setgid dans le système de fichier.

Composants

- La librairie cmap est utilisée pour interagir avec la base de configuration utilisée par corosync.
- La librairie cpg est utilisée pour créer des applications distribuées qui opèrent proprement durant le partitionement, fusions, et pannes du cluster.
- la librairie sam fournis un outils pour vérifier l'état de santé d'une application. Son but est de redémarrer un processus local quand il plante pour répondre à un requête d'état de santé dans un intervalle de temps configuré.
- La librairie quorum est chargé dans tous les nœuds du cluster et track le statut du quorum d'un nœud. Pour que ce service soit utile, un fournisseur de quorum doit être configuré.
- La librairie votequorum est optionnellement chargé dans tous les nœuds dans le cluster pour éviter les situations de split-brain. Il permet cela en ayant un nombre de votes assignés à chaque système dans le cluster et s'assurer que seulement lorsqu'une majorité de votes sont présents, les opérations du cluster sont autorisés à être traités.