

Gestion de plusieurs instances OSPF avec les contextes SNMP

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Informations générales](#)

[Snm context averti](#)

[Configuration](#)

[Vérifiez](#)

[Vérification SNMPv2](#)

[Vérification SNMPv3](#)

[Informations connexes](#)

[Introduction](#)

Ce document fournit des configurations d'échantillon pour SNMPv2 et SNMPv3 qui décrivent comment employer des snmps context pour gérer des multiples instances de Protocole OSPF (Open Shortest Path First).

[Conditions préalables](#)

[Conditions requises](#)

Aucune spécification déterminée n'est requise pour ce document.

[Composants utilisés](#)

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

[Conventions](#)

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à

Informations générales

Le MIB OSPF défini par l'IETF ([RFC 1850](#)) a été conçu pour fonctionner avec seulement un processus/exemple OSPF sur un routeur donné.

Par exemple, il y a seulement un objet simple d'*ospfRouterId*, pas une table de eux. Afin de manipuler des multiples instances, [RFC 4750](#) suggère que vous employiez les contextes SNMPv3 pour fournir des vues de par-exemple.

Snmp context averti

Avant de mettre au courant le contexte de code SNMP OSPF IOS, le système sélectionnerait l'exemple « par défaut » plus ou moins aléatoire quand il a renvoyé les objets scalaires et quelques tables. Dans des ces cas, les informations des autres exemples n'étaient pas disponibles par l'intermédiaire du SNMP. Pour quelques autres tables, le SNMP écraserait ensemble les entrées de tous les exemples sans n'importe quelle manière de discerner ce qui était quel. Dans de nombreux cas, ceci a pu mener à ambigu ou aux entrées en double. Ce n'était pas particulièrement bonne pratique dans des configurations PE-CE où les adresses IP et les router-id de voisin ne pourraient pas être seuls. Ceci a fait la surveillance et le CE individuel de dépannage cite difficile ou impossible.

Avec le code contexte-averti en cours IOS (quand aucun contexte n'est spécifié), le vieux comportement pour les objets scalaires existe toujours. La seule modification est qu'elle limite maintenant tous plutôt que juste certaines des tables au même exemple « par défaut » OSPF que les grandeurs scalaires. Quand des contextes sont fournis, les requêtes SNMP peuvent être visées à un exemple particulier OSPF, et toutes les informations pour cet exemple peuvent être récupérées d'une manière cohérente et non ambiguë.

Si SNMPv3 est utilisé, la chaîne de contexte peut être fournie directement avec le balayage. Le SNMPv2C ne fournit pas un cadre. Cependant, vous pouvez tracer des chaînes de caractères de la communauté SNMP aux contextes dans la configuration IOS, et ces contextes peuvent être utilisés pour diriger les balayages SNMPv2 vers un exemple OSPF de particularité.

Configuration

Cet exemple de configuration est basé sur SNMPv2 :

```
Router 1
Router1#
router ospf 1
  router-id 1.1.1.111
  log-adjacency-changes
  snmp context context1
!
router ospf 2
  router-id 4.4.4.111
  log-adjacency-changes
  snmp context context2
!--- Associates the SNMP context with the instance. !
```

```
snmp-server user u2 g2 v2c !--- Configures the user u2
to the SNMP group g2 and !--- specifies the group is
using the SNMPv2c security model. snmp-server group g2
v2c !--- Configures the SNMP group g2 and specifies !---
the group is using the SNMPv2c security model. snmp-
server group g2 v2c context context1 snmp-server group
g2 v2c context context2 snmp-server community public RO
!--- Community access string to permit access !--- to
the SNMP. snmp-server community cx1 RO snmp-server
community cx2 RO snmp-server context context1 snmp-
server context context2 snmp mib community-map cx1
context context1 security-name u2 !--- Associates the
SNMP community cx1 with !--- the context context 1. snmp
mib community-map cx2 context context2 security-name u2
```

Cet exemple de configuration est basé sur SNMPv3 :

Routeur 1

```
Router1#
router ospf 1
  router-id 1.1.1.111
  log-adjacency-changes
  snmp context context1
!
router ospf 2
  router-id 4.4.4.111
  log-adjacency-changes
  snmp context context2
!
snmp-server user u1 g1 v3
snmp-server group g1 v3 noauth
snmp-server group g1 v3 noauth context context1
snmp-server group g1 v3 noauth context context2
snmp-server context context1
snmp-server context context2
```

Remarque: Utilisez l'outil [Command Lookup Tool](#) (clients [enregistrés](#) seulement) pour trouver plus d'informations sur les commandes utilisées dans ce document.

Vérifiez

Vous pouvez employer la commande de **snmpwalk** sur n'importe quelle machine cliente afin de vérifier la sortie.

Remarque: L'[Outil Interpréteur de sortie](#) (clients [enregistrés](#) uniquement) (OIT) prend en charge certaines commandes **show**. Utilisez l'OIT pour afficher une analyse de la sortie de la commande **show**.

Vérification SNMPv2

SNMPv2

```
linux>snmpwalk -c public -v2c irp-view14:7890 OSPF-
MIB::ospfRouterId.0 OSPF-MIB::ospfRouterId.0 =
IpAddress: 4.4.4.111 linux>snmpwalk -c cx1 -v2c irp-
view14:7890 OSPF-MIB::ospfRouterId.0 OSPF-
MIB::ospfRouterId.0 = IpAddress: 1.1.1.111
```

```
linux>snmpwalk -c cx2 -v2c irp-view14:7890 OSPF-
MIB::ospfRouterId.0 OSPF-MIB::ospfRouterId.0 =
IpAddress: 4.4.4.111
```

Vérification SNMPv3

SNMPv3

```
linux>snmpwalk -u u1 -v3 irp-view14:7890 OSPF-
MIB::ospfRouterId.0 OSPF-MIB::ospfRouterId.0 =
IpAddress: 4.4.4.111 linux>snmpwalk -u u1 -v3 -n
context1 irp-view14:7890 OSPF-MIB::ospfRouterId.0 OSPF-
MIB::ospfRouterId.0 = IpAddress: 1.1.1.111
linux>snmpwalk -u u1 -v3 -n context2 irp-view14:7890
OSPF-MIB::ospfRouterId.0 OSPF-MIB::ospfRouterId.0 =
IpAddress: 4.4.4.111
```

Informations connexes

- [Gestion de la configuration OSPF avec SNMP](#)
- [Support et documentation techniques - Cisco Systems](#)