Surveillance continue par SNMP d'un réseau

Daniel Guéniche
gueniche@polycnrs-gre.fr

2 octobre 1995

1 Présentation:

Au Polygone CNRS de Grenoble nous avons un réseau en étoile. Devant chaque entité (laboratoire ou service) un routeur sépare le trafic du *backbone* des trafics internes (voir schéma).

Une station dite *centrale* (peut importe son emplacement réel):

- contrôle en permanence le bon fonctionnement des 11 routeurs [1],
- s'assure de la disponibilité du *backbone* et de l'accès extérieur,

Une machine dite *station-sonde* dans chaque entité importante est configurée pour contrôler régulièrement les répéteurs et s'assurer de la disponibilité du réseau. Elle ne remonte à la station centrale que les anomalies. Cette répartition accroît la fiabilité de l'ensemble, limite pour la machine centrale le nombre d'équipements à interroger, et réparti le trafic *snmp*.

Les problèmes sont signalés à l'aide de messages **vocaux** reconstitués (le *mail* n'est pas satisfaisant car nous en sommes déjà inondés).

Le tout s'appuie sur des briques simples et fiables telles que *snmp* ou *ping*. Dès leur mise en place ces programmes effectuent une véritable radioscopie d'un réseau. Puis garantissent sa qualité opérationnelle.

2 But:

- Etre avisé au plus tôt du moindre incident survenant sur un réseau même étendu par des messages vocaux que vous pré-composez.
- Ne pas avoir à fixer un écran pour ne pas aggraver la fatigue visuelle.
- Pouvoir vaccer l'esprit tranquille à d'autres occupations.
- Pouvoir programmer des réactions conditionnelles.
- N'avoir aucun logiciel à acheter.

[1] notés Rt_xxxxx sur le schéma

3 Configuration minimum, Administrateurs concernés:

La station centralisant la surveillance doit être équipée d'un haut-parleur et d'un micro. Seule la période de collecte des informations *snmp* accapare son CPU.

Des agents *snmp* (routeurs, ponts ou répéteurs) doivent avoir été placés en entrée des diverses entités composant le réseau (laboratoires, bâtiments, étages, ...) et au sein de ces entités si celles-ci sont conséquentes.

4 Principe:

Sur la station centrale:

- Un programme (**Rts_look**.c) interroge régulièrement en *snmp* les équipements de premier niveau (des routeurs au Polygone CNRS).
- Un shell (**check**) garanti son bon fonctionnement et effectue des contrôles complémentaires.

Au sein de chaque entité:

- **Rptr_look**.c interroge régulièrement les équipements de second niveau (des répéteurs au Polygone CNRS), et remonte les incidents.
- **check** le supervise, effectue des tests complémentaires, avise le responsable local de toute anomalie.

Rts_look toutes les *n* minutes:

- parcourt un fichier de commandes (**Rts_base**). Chaque ligne contient le nom du routeur à surveiller, sa marque [2], éventuellement la *community string* [3], les interfaces à interroger et les valeurs de la derniere collecte (nombre d'octets transmis, reçus, nombre de collisions, d'erreurs, etc.)
- émet l'ordre *snmp* et signale si l'équipement ne répond pas ou si la requête a échoué.
- compare les valeurs collectées avec celles précédemment enregistrées. Des seuils paramétrables (en Rts_look.h) déterminent s'il faut signaler une anomalie simplement oralement et dans la trace générale, ou s'il faut aussi envoyer un *mail* au responsable.
- sauve les valeurs de la nouvelle collecte en Rts_base.

Rptr_look effectue le même travail sur les répéteurs.

check tourne aussi en tâche de fond, sur la station centrale toutes les *n* minutes:

- il s'assure que Rts_look tourne,

[2] La requête *snmp* peut varier d'un équipement à l'autre (nous avons du Xyplex, du CISCO et du 3Com)

[3] Chaine de caractère authentifiant une requète *snmp*.

- si la station a reçu un *trap snmp* [4] d'un équipement (routeurs, passerelles Farallon ou Kinétics, répéteurs, etc.), ou un message d'une sonde, ou un *logging trap* d'un routeur CISCO [5], il appelle le programme correspondant qui:
  - décodera le message,
  - avisera éventuellement le responsable de l'entité concernée,
  - constituera et émettra un message vocal (*play*),
  - déposera un message dans la trace générale.
- il teste par un *ping* l'accessibilité du routeur de site (Rt_CNRS sur le schéma) et des routeurs de France Télécom (Rt_FT) à chaque bout de notre ligne spécialisée. Les resultats de ces *ping* d'une trame de 512 bytes est précieusement stocké car:
  1. ils seront utilisés durant la nuit pour préparer les courbes de réponse en ces divers points pour la journée écoulée.
  2. ils sont immédiatement exploités par un programme graphique qui affiche en continu la disponibilité de la ligne.
  3. ils permettent de s'assurer du bon fonctionnement du resolver de noms.

Sur les stations-sonde, check surveille Rptr_look, fait remonter les incidents et vérifie l'accessibilité d'une liste d'adresses (envoie directement un *mail* au responsable en cas de problème).

Enfin un shell (**CHECK**) activé par *cron* toutes les heures s'assure de la présence de check lui-même et que Rts_look travaille effectivement.

5 Tests instantanés:

Rt_look (sans **s**) sert à interroger "manuellement" un équipement. Il lance le même ordre *snmp* que Rts_look et compare ses résultats avec les dernières valeurs relevées par ce dernier. On peut ainsi suivre plus finement une dérive d'un équipement. Si l'argument "8h" est fourni, la comparaison prend pour référence les valeurs du matin à 8 heures.

6 Installation:

Vous trouverez en *ftp anonymous* sur labs.polycnrs-gre.fr en /pub/chambery un package **Net_look.tar** mis à la disposition de la communauté qui contient:

- un README (étapes d'installation)
- un package *SNMP* minimum (immédiatement utilisable sur SUN OS 4.1.2)
- les programmes C, les shells, un exemple de **base**

[4] Remontée d'alerte à l'initiative de l'équipement (réinitialisation, lien qui tombe, ...)
[5] Propre à CISCO: l'équipement signale de lui-même les collisions excessives, la perturbation d'un brin, ...

7 Notes:

Vous pouvez recevoir l'avis suivant:

```
>SNMP>  20Sep95  07h45mn:  200  % de colls cote port8  du Rptr-LN.
```

Plus de 100% d'erreurs ou de collisions peuvent surprendre. Le pourcentage est pourtant la façon la plus immédiatement lisible pour renseigner sur la proportion de paquets ayant pu être émis ou reçus sans problème, et ceux ayant entrainé des collisions ou des erreurs. De la même façon, 100% de paquets en erreurs ne signifie pas qu'aucun n'a pu être reçu, mais qu'il y a eu autant de paquets en erreurs que de paquets sains.