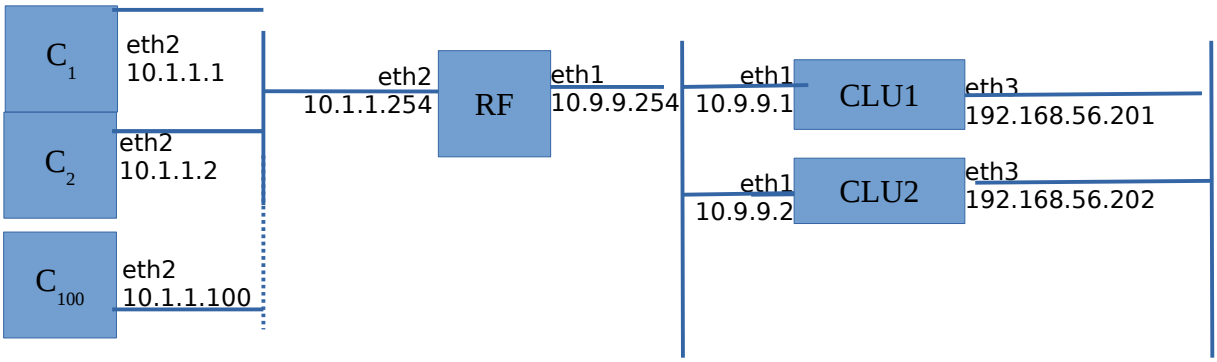
Laboratorio di Amministrazione di Sistemi T

Prova pratica del 5 novembre 2019

Descrizione generale del problema

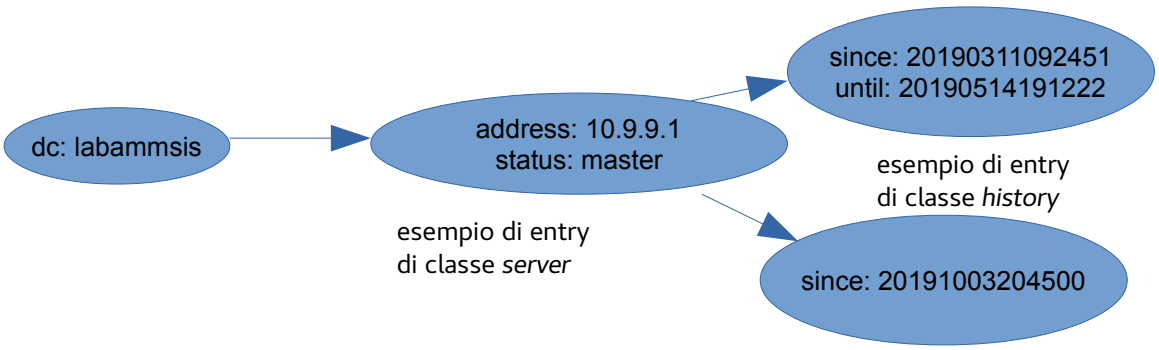


Un sistema di logging ad alta disponibilità è formato da due macchine server (CLU1, CLU2).
Ogni server verifica via SNMP lo stato dell'altro ogni 10 secondi, e si dichiara master se per un minuto consecutivo l'altro non è raggiungibile o non ha il processo *rsyslogd* in ascolto sulla porta UDP 514.
In caso contrario un server deve considerarsi in standby.
Quando un server standby diventa master, deve accertarsi di avere il demone rsyslog attivo.
Quando un server master diventa standby, deve accertarsi di avere il demone rsyslog spento.

Il router-firewall (RF) media le richieste di servizio tra i client ed il server master; i client non sanno quale sia il master, quindi dialogano solo con RF, che procederà a inoltrare per mezzo del packet filter il traffico al server giusto.

Si noti che quando un client invia un messaggio di log, questo verrà scritto solo sul server master.

RF ospita una directory LDAP che tiene traccia dei periodi in cui ogni server assume il ruolo master, organizzando i dati come da esempio di DIT qui riportato:



Le entry di classe *server* elencano i server e mantengono nell'attributo *status* lo stato corrente (master o standby). Le entry di classe *history* relative ad un server memorizzano ognuna un periodo di tempo in cui il server è stato master; l'attributo *since* indica l'inizio del periodo, l'attributo *until* la fine (non è presente nel caso il server sia diventato master e non ancora tornato slave).

File da consegnare

(tra parentesi quadre la collocazione: C = disponibile su tutti i client / R = disponibile sul router / S = disponibile su tutti i server; parentesi e collocazione non fanno parte del nome)

svc.schema.ldif [R] – Definire i tipi di attributo *address* e *status* (stringhe), *since* e *until* (interi), e le classi *server* e *history* che li utilizzino come descritto.

server.sh [S] - Questo script implementa continuamente i controlli per decidere se il server su cui è in esecuzione è "master" o "standby" come descritto nella sezione evidenziata della pagina precedente; il controllo via SNMP deve essere fatto su entrambe le reti che interconnettono i server, è sufficiente ricevere risposta da una delle due interrogazioni per considerare l'altro server raggiungibile.
Quando lo stato cambia, lo script:

- agisce perché lo stato del servizio rsyslog sia quello corretto
- invoca *ldaptool.sh* passando come parametri l'indirizzo del server e il nuovo stato

Indicare nei commenti come configurare l'agent SNMP dei server per consentire la rilevazione della presenza di rsyslogd in ascolto sulla porta UDP 514.

ldaptool.sh [S] – Questo script richiede due parametri sulla riga di comando: l'indirizzo di un server e una stringa che deve valere "master" o "standby".
In funzione dei valori ricevuti aggiorna l'attributo *status* della entry di classe *server* corrispondente.
Se invocato con "master", deve creare, al di sotto della entry relativa al server, una nuova entry di classe *history* inizializzando *since* al tempo corrente.
Se invocato con "standby", deve individuare, al di sotto della entry relativa al server, la entry di classe *history* in cui manca *until* e introdurvelo impostandolo al tempo corrente.
Il tempo corrente deve essere espresso nel formato *YYYYMMDDhhmmss* (sostituendo le lettere con anno mese giorno ora minuto e secondo)

reroute.sh [R] – Questo script osserva senza sosta il traffico in entrata sulla porta di LDAP. Quando rileva un pacchetto che indica la fine di una connessione, legge dalla directory qual è il server master e configura il packet filter per dirigere verso di esso le connessioni *ssh* e *syslog* provenienti dalla rete dei client e dirette al router.

logrotate.sh [S] – Questo script rinomina il file **`/var/log/user.log`** in **`/var/log/user.YYYYMMDDhh`** (le lettere indicano anno mese giorno e ora dell'istante di esecuzione) e riavvia il servizio rsyslog per garantire che venga messo a riposo.
Indicare nei commenti

- come configurare rsyslog sui client e sui server perché i messaggi etichettati local1.info prodotti sui client vengano scritti su **`/var/log/user.log`** del server master (tenendo conto di come opera *reroute.sh*)
- come far eseguire automaticamente lo script all'inizio di ogni ora

getlog.sh [R] – Questo script può essere invocato via ssh da un client, passando come parametro un timestamp nello stesso formato usato da *ldaptool.sh*.
Individua quale server era master in quel momento, recupera da esso via ssh il contenuto del relativo file di log, e lo produce su stdout in modo che venga trasferito fino al client che ha invocato lo script.
Indicare nei commenti come predisporre i sistemi in modo che i trasferimenti ssh non richiedano password.