Uitgebreid voorstel Masterproef Informatica

Titel van het project : Evaluatie van de schaalbaarheid van SNMP-gebaseerde netwerkbevraging

Datum indienen : 04/01/2014

Naam student : Daan Volcke

Interne promotor: Joris Moreau

In samenwerking met : Bedrijf en IMinds

Algemene informatie voor extern bedrijf:

Naam van het bedrijf : NetworkMining
Adres: Paleizenstraat 44, 1030 Brussel

- Is dit de 1e masterproef in het bedrijf in samenwerking met onze opleiding? **Ja**/Nee
- Is er in het bedrijf inhoudelijke en technische begeleiding mogelijk ? **Ja**/Nee
- Kan de student in het tweede semester (februari-mei) 3 dagen per week in het bedrijf/onderzoekscentrum aanwezig zijn om te werken aan de masterproef? **Ja**/Nee

Begeleiding :

Externe promoteren: Didier Colle, Nico Wauters
Andere begeleiders: Wouter Tavernier, Leo Nederlof

Bespreking door de werkgroep (niet invullen bij indienen van een voorstel)

Beslissing: goedgekeurd - herwerken tegen … / …
Minimale uitbreidingen:
Opmerkingen:
Advies van collega's:

Doelstelling van het project :

Bestaande (open-source) softwaretools voor het opvragen van netwerkinformatie via SNMP beschikken over te weinig intelligentie om dit mogelijk te maken voor grootschalige netwerken met meer dan 1000 netwerkelementen. Hierdoor is het moeilijk om op frequente basis datamining van het netwerk te doen om bijvoorbeeld de netwerkconnectiviteit of netwerkroutering in kaart te brengen. Het efficiënt nagaan van configuratiefouten van grootschalige Ethernet- en IP-netwerken is nochtans een must voor de netwerkbeheerder.

De bedoeling van de thesis bestaat erin om een stuk software te schrijven dat in staat is om grootschalige netwerken van routers en switches op efficiënte manier te bevragen via hun SNMP-agent.

Bestaande situatie en probleemstelling :

Fabrikanten van routers en switches voorzien nu al network management systemen die het leven van een netwerkbeheerder makkelijker maken. Deze systemen kunnen onder andere geaggregeerde informatie van de verschillende netwerkelementen rapporteren. Het probleem hierbij echter is dat deze management systemen enkel werken voor netwerkelementen van dezelfde fabrikant. Grootschalige netwerken bestaan echter uit apparatuur van verschillende fabrikanten, typisch bepaald door het afwegen van de kostprijs en features die een bepaalde fabrikant aanbiedt op het moment dat een netwerk uitgebreid wordt. Elke fabrikant biedt zo wel een network management systeem aan voor de eigen apparaten, maar ook het raadplegen van deze systemen is ook niet uniform: er wordt gebruik gemaakt van verschillende API's en technologieën zoals XML SOAP. Ze bieden ook niet altijd alle informatie aan die de netwerkbeheerder wenst. Vandaar dat ervoor geopteerd werd om gebruik te maken van het SNMP protocol voor het opvragen van netwerkinformatie. Dit protocol wordt wel door apparatuur van alle fabrikanten ondersteund en biedt een enigszins uniform alternatief. Het SNMP protocol biedt niet dezelfde aggregatiemogelijkheden als network management systemen. In de plaats daarvan gaat het over ruwe informatie van individuele netwerkcomponenten die verder verwerkt en geaggregeerd moet worden.

Op dit moment wordt er netwerkbevraging gedaan via SNMP met behulp van een zelf ontwikkelde tool. Deze beschikt over nog niet veel intelligentie, momenteel overloopt ze simpelweg iteratief alle netwerkcomponenten en vraagt de relevante SNMP informatie op. Er is ook nog geen sprake van het bijhouden van historische data. Het is duidelijk dat er nog veel plaats is voor verbeteringen: multithreading voorzien, gedistribueerde werking, enige intelligentie bij de opvragingen en historische data bijhouden. De tool wordt momenteel slechts gebruikt op kleine schaal waarbij performantie nog niet van groot belang is. Er zal dus nog moeten onderzocht worden hoever er zal moeten geoptimaliseerd worden om de tool ook op grootschalige netwerken te kunnen inzetten.

Gedetailleerde omschrijving van de opdracht die minimaal moet worden verwezenlijkt:

Op zijn minst moet de schaalbaarheid van SNMP opvragingen met de huidig gebruikte software onderzocht worden. Hierbij moet er gezocht worden naar mogelijke bottlenecks die zich voordoen. Dit kan gaan om de CPU van de client, bandbreedteproblemen, de databank die de opgevraagde gegevens moet opslaan die niet kan volgen of het netwerkelement zelf die niet snel genoeg is.
Eens de bottlenecks geïdentificeerd zijn moet er gezocht worden naar oplossingen om de bottlenecks te verhelpen. Denk aan aanpassingen aan de SNMP retriever zoals het implementeren van multithreading, gelijktijdig gebruik van meerdere SNMP clients of het opzetten van een databankcluster.
Om te zien hoe effectief de oplossingen zijn zal er ook een testmethode/benchmark opgesteld moeten worden om dit na te meten.

Om een idee te krijgen van de huidige performantie van de SNMP retriever werd er een klein testnetwerk opgezet met een tiental Linuxmachines en werd van alle machines SNMP informatie opgevraagd. Het heeft maar liefst acht uur geduurd eer de retriever klaar was met alle gegevens op te halen! Het is duidelijk dat deze tijd met een veelvoud omlaag moet kunnen gehaald worden en er dus

nog zat ruimte is om de software te optimaliseren. Een goed startpunt van de thesis zou dan ook het onderzoeken van de SNMP retriever zijn, om aan te tonen of er al dan niet een bottleneck in aanwezig is. De SNMP retriever is ook de meest waarschijnlijke plek voor mogelijke bottlenecks: we weten dat de huidige versie van de SNMP retriever een simpel stuk software is met weinig intelligentie of optimalisaties. De netwerkapparatuur is voorzien om SNMP requests af te handelen en zou dan ook uitvoerig moeten getest geweest zijn door de fabrikant. Ook het netwerk zou geen problemen mogen hebben met SNMP requests van een 20-tal bytes per pakket. Althans niet zolang de SNMP retriever sequentieel alle netwerkcomponenten overloopt en sequentieel de SNMP gegevens opvraagt. Wanneer er multithreading of parallelle SNMP retrievers geïntroduceerd worden zou daar verandering in kunnen komen.
Vermits de beperkingen van de huidige SNMP retriever al tot uitdrukking komen op kleine netwerken betekent dit ook dat er geen afhankelijkheid is van een groot netwerk om het effect te testen van de geïmplementeerde oplossingen tegen de bottlenecks in de SNMP retriever.

Gezien de performantie veronderstelden we dat de huidige SNMP retriever gebruik maakt van SNMP walk opdrachten om de gegevens om te vragen. Deze vermoedens werden ook bevestigd bij navraag. Bij SNMP walk opdrachten wordt de OID van een gegeven meegegeven aan het SNMP request en wordt er gewacht op het antwoord eer het volgende gegeven wordt opgevraagd. Een SNMP bericht is slechts ongeveer 20 bytes groot, het zou veel beter zijn om meteen de SNMP antwoorden zo vol mogelijk te steken met de gegevens van meerdere OID's. Dit kan gebeuren met behulp van SNMP GetBulkRequest operaties in SNMPv2. Die voert meerdere SNMP getnext operaties uit en stopt de antwoorden samen in een pakket in plaats van voor elke getnext operatie een apart pakket.
Om een idee te krijgen van de mogelijke theoretische tijdswinst die hiermee kan geboekt worden: voor elk OID wordt een apart netwerkpakket verstuurd. De netwerkvertraging wordt gemeten in milliseconden. Dus voor elke OID is er een vertraging van enkele ms. Als we meteen de eerstvolgende 100 OID's met een SNMP GetBulkRequest opvragen, dan gaat het om een slechts een pakket. Stel een netwerkvertraging van 50ms, voor 100 OID's duurt dit met een SNMP walk 5 seconden, met een GetBulkRequest slechts 50ms, dus een factor 100. Hoe meer OID's in een pakket, hoe beter.
De broncode zal ter beschikking gesteld worden zodat de huidige werking kan geanalyseerd worden en SNMP GetBulkRequests geïmplementeerd kunnen worden.

Problemen die moeten opgelost worden (niet te gedetailleerd):

Zoals hierboven uitgelegd moeten de geïdentificeerde bottlenecks verholpen worden. Waarna moet nagegaan worden hoe effectief de oplossingen werken door middel van een testmethode of benchmark.

Technologieën die aan bod komen :

De belangrijkste technologie die gebruikt zal worden is natuurlijk SNMP. Hiermee kunnen we op uniforme wijze de gegevens opvragen van alle netwerkcomponenten, zonder rekening te hoeven houden met de fabrikant van het toestel.
Verder is de softwaretool die momenteel gebruikt wordt om op kleine schaal de SNMP opvragingen te doen geschreven in Visual Basic.

Mogelijke uitbreidingen en opties :

Als uitbreiding kan de bestaande software voorzien worden van enige intelligentie bij het opvragen van netwerkinformatie. Het is zo dat sommige gegevens zeer dynamisch zijn en andersom er gegevens zijn die quasi statisch zijn. Denk bijvoorbeeld aan temperatuurmetingen en een overzicht van alle netwerkinterfaces die aanwezig zijn in een toestel. Het eerste gegeven verandert constant, het laatste haast nooit. Het is dan ook logisch dat het laatste niet zo vaak opgevraagd moet worden als het eerste. Zo zouden we kunnen een algoritme of heuristiek ontwikkelen die de veranderlijkheid van gegevens bepaalt en daaruit beslist hoe vaak dat gegeven opgehaald moet worden.
Sommige gegevens kunnen meer of minder belangrijk zijn als andere. De netwerkbeheerder zou dan ook zelf kunnen de periodiciteit opgeven voor het opvragen van bepaalde gegevens afhankelijk van het belang dat de netwerkbeheerder eraan hecht. Dit is ook een interessante optie bij het bijhouden van historische data.

Een ander idee is om te zien of de netwerkcomponenten SNMP requests beantwoorden die via broadcast of multicast (na het inschrijven op een multicastgroep) verstuurd zijn geweest. Bij Linux machines is dit bijvoorbeeld wel het geval. Hierdoor zou het netwerkverkeer om SNMP informatie te verzamelen quasi gehalveerd worden. De SNMP retriever moet niet meer elk netwerkelement individueel ondervragen maar ondervraagt ze allemaal tegelijkertijd. Omgekeerd blijft het netwerkvolume wel even groot, en afhankelijk van de omvang van het netwerk zou dit ook wel eens een zeer zware belasting voor de retriever kunnen zijn. De SNMP retriever moet dan tenslotte de antwoorden van alle netwerkelementen tegelijkertijd verwerken. Toch is het zeker een optie die de moeite waard is om te onderzoeken. Er moet dan ook gekeken worden wat voor aanpassingen er nodig zijn aan het netwerk om dit te ondersteunen: het inschrijven van de nodes op een multicastgroep, het toestaan van het routeren van multicastverkeer over de routers heen en bijhorende (eventueel statische) multicastroutes.

Vernieuwende aspecten :

In het vak Computernetwerken III is er reeds kennis gemaakt met de theoretische achtergrond van SNMP. Aan de praktische kant werden demonstraties gegeven van het opvragen van gegevens via SNMP en het gebruik van een MIB browser. Tenslotte werd er ook geëxperimenteerd met het manueel opstellen van SNMP requests via een hex editor.
In de opleiding werden de problemen aangekaart bij het opvragen van ganse verzamelingen data met SNMP walks. Het probleem hierbij is dat er achter de schermen voor ieder gegeven een apart SNMP request verstuurd wordt. Het is met andere woorden niet mogelijk met SNMP om gegevens in bulk op te vragen. Dit kan al bij een enkele host al een probleem vormen, maar het probleem kan dramatische proporties aannemen als het aantal op te vragen gegevens toeneemt in een grootschalig netwerk. De tweede versie van SNMP biedt wel de mogelijkheid om gegevens in bulk op te vragen.

Alhoewel broadcast- en multicastverkeer wel uitvoerig aan bod is gekomen in het vak computernetwerken, is dat niet het geval voor de combinatie met SNMP bevragingen. Dit gedrag kan verschillen van fabrikant tot fabrikant, en zelfs van model tot model. Bij een aantal Linuxmachines wordt dit bijvoorbeeld wel ondersteund.

Ook het opsporen van bottlenecks in software, op toestellen, op het netwerk of op een databank zijn aspecten die weinig aan bod gekomen zijn in de opleiding.