

Practicum 3

Doel

- Uitwerking van de “node life cycle” van het ringnetwerk.
 - Discovery
 - Bootstrap
 - Shutdown
 - Failure

Opdrachten

- Discovery: ontdekken van de nameserver en bestaande nodes in het ringnetwerk
- Bootstrap: opstarten van de node, de initialisatie van de lokale parameters en het aanpassen van de parameters in de bestaande nodes
- Shutdown: Het verlaten van het ringnetwerk en het updaten van de parameters in de buurnodes en de nameserver
- Failure: Het detecteren van een node die faalt en het updaten van de parameters in de buurnodes en de nameserver

Specificatie

!!! Je mag er van uit gaan dat er een Nameserver in het lokale netwerk bestaat en dat deze bereikbaar is via multicast!!!

OPSTARTEN (BOOTSTRAP + DISCOVERY)

- 1) Ontwikkel eerst een methode die via een multicast bericht de bestaande nodes en de nameserver bereikt; multicast wordt o.a. gebruikt in stap 3
- 2) Ontwikkel een methode die op basis van de nodenaam een hash kan berekenen (deze methode zou je reeds moeten hebben van de vorige sessie)
- 3) Verstuur bij opstart van een node de naam en het IP van deze node via de multicast methode uit stap 1 naar de nodes en de nameserver in het netwerk.
- 4) De nameserver ontvangt het bericht en doet volgende stappen:
 - a. Bereken hash van de nodenaam
 - b. Plaats Hash + IP in de Map (zie vorige sessie)
 - c. Antwoordt aan de opstartende node met het aantal nodes dat zich momenteel in het netwerk bevinden.
- 5) De andere nodes in het netwerk ontvangen het bericht en doen het volgende:
 - a. Bereken de hash
 - b. Indien de huidige id < hash < volgende id *//ik ben de vorige node...*
 - i. Update volgende id met de hash van de nieuwe node
 - ii. Antwoordt aan de opstartende node met de originele huidige en volgende id
 - c. Indien de vorige id < hash < huidige id *//ik ben de volgende node...*
 - i. Update vorige id met hash
- 6) De huidige node ontvangt een antwoord van de nameserver en van de voorgaande node en doet het volgende:
 - a. Als aantal nodes < 1 *//ik ben de 1^{ste} node...*
 - i. Volgende node = huidige node
 - ii. Vorige node = huidige node

- b. Aantal nodes ≥ 1 *//ik plaats me tussen vorige en volgende node...*
 - i. Volgende node = volgende node ontvangen van vorige node
 - ii. Vorige node = vorige node ontvangen van vorige node
- c. Initialiseer vorige id en volgende id met info van de voorgaande node

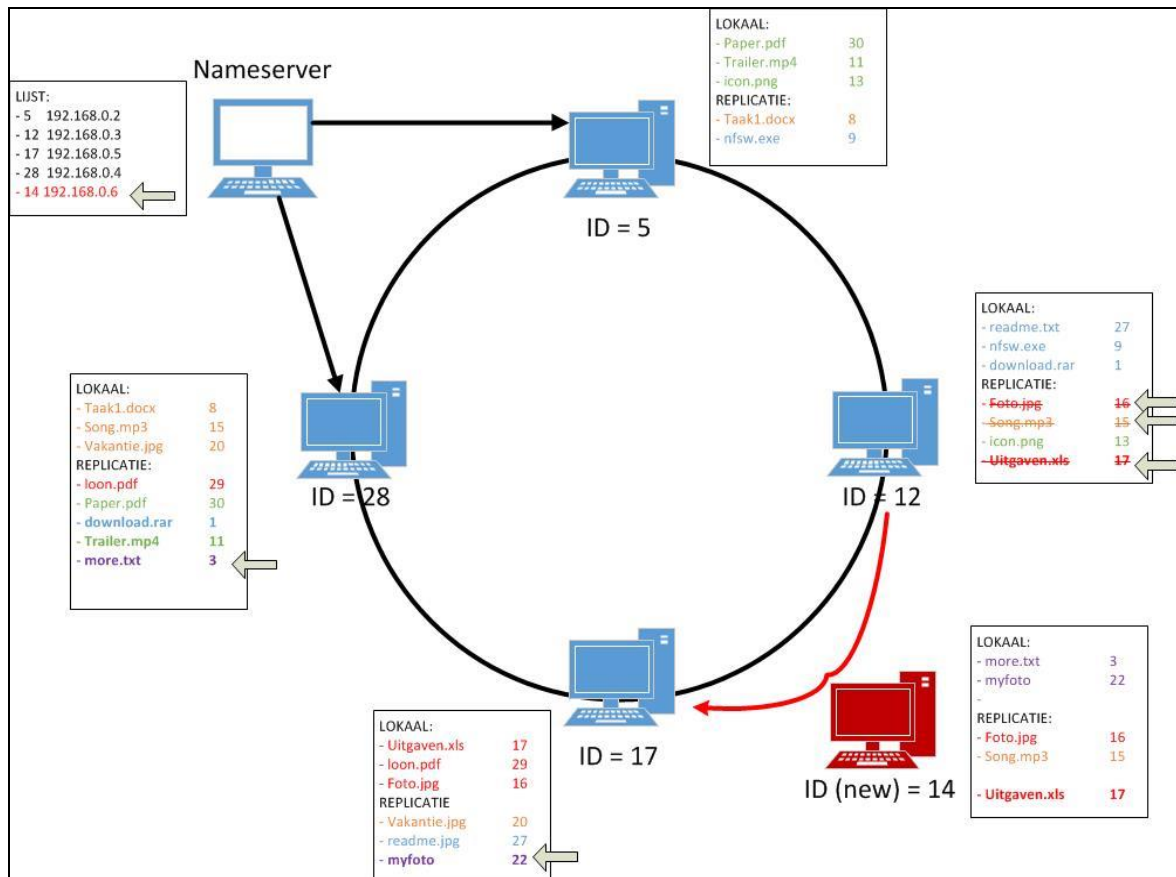
SHUTDOWN

- 1) Stuur de id van de volgende node door naar de vorige node
- 2) In de vorige node wordt de volgende node aangepast met deze info
- 3) Stuur de id van de vorige node op naar de volgende node
- 4) In de volgende node wordt de vorige node aangepast met deze info
- 5) Verwijder de node bij de nameserver
- 6) Stop het System Y proces

FAILURE

- 1) Deze methode moet opgeroepen worden in elke exception in elke node die een communicatie doet met een andere node. Op deze manier wordt er gedistribueerd gedetecteerd of een node faalt.
- 2) Vraag bij de nameserver de vorige en volgende node op van de node die faalt
- 3) Update bij de vorige node de volgende node met de info die je van de nameserver krijgt
- 4) Update bij de volgende node de vorige node met de info die je van de nameserver krijgt
- 5) Verwijder de node bij de nameserver.
- 6) Test deze methode met een ping methode die je bij elke node kan oproepen gecombineerd met het killen (ctrl – c) van de node.

In figuur 3 zien we een situatie waarbij node met id = 14 zich plaatst tussen node id=12 en id=17; de bestandsreplicatie informatie (aangeduid pijltjes) zullen we voorlopig negeren.



Figuur 3, nieuwe node, met ID = 14 nestelt zich tussen node ID 12 en ID 17

Testen

Zorg er voor dat er testen voorzien zijn voor:

- Bootstrap
- Shutdown
- Failure

Zorg er voor dat de status van elk van de componenten in deze test afgetoetst wordt:

- id prev (elke node)
- id next (elke node)
- nameserver tabel
- ...

Steek zoveel mogelijk van de test in programmeercode en voeg eventuele extra handelingen zoals het uittrekken van een UTP kabel toe in een document die de testen beschrijft.