

클러스터 기반의 웹-메일 시스템의 새로운 구조

강호석⁰, 박은영, 심영철
홍익대학교 컴퓨터공학과

{hskang, eypark, shim}@cs.hongik.ac.kr

A New Architecture for Cluster-Based Web-Mail

Ho-Seok Kang⁰, En-Young Park, Young-Chul Shim
Hong-Ik University, Dept. of Computer Engineering

요 약

많은 네트워크 서비스들 중에 웹 서비스와 더불어 가장 많이 사용하고 있는 전자메일 시스템은 급속한 성장으로 클러스터 기반 환경으로 이용하는 것이 효율적이다. 이런 클러스터 기반의 메일 시스템 중 Earthlink는 부족한 부하관리와 결합허용 능력을 가지고 있고, Porcupine은 지나친 부하균등으로 인한 시스템의 복잡성을 가지고 있다. 이 두 시스템의 균형을 이루게 하고 여기에 웹 인터페이스를 적용한 웹 메일 시스템을 사용하여 사용자 중심의 동작 시나리오를 구성하고 부하관리와 결합허용 능력을 유지하는 구조를 제안하였다.

제안한 클러스터 기반의 메일 시스템에서 특수한 상황으로 발생하는 공지, 전자 잡지 배송등 내부 전체 사용자나 일부 사용자에게 메일을 보내는 상황을 속도와 저장공간 측면에서 효율적으로 적용하는 구조를 제안하였다.

른 문제 그리고 일부 사용자만 많은 메일을 사용할 경우에 발생하는 파일서버간의 부하의 불균형을 발생시킬 수 있다[1].

1. 서론

네트워크 환경이 발달하면서 이러한 기반 환경을 이용한 서비스중 하나인 전자우편의 사용은 일반인에게도 필수적인 사항이 되었다. 많은 수의 사람이 전자우편을 사용하므로 기존의 전자우편 시스템 보다 고성능의 시스템이 요구된다. 이를 위해 고안된 것이 최근 여러 곳에서 개발되고 있는 클러스터링 방법을 이용해서 전자우편 시스템을 만들어보고자 하는 것이다. 이를 이용해서 기존의 전자우편 시스템의 약점들을 보완하고 좀 더 편리한 시스템을 개발하고자 하는 목적이 있다.

기존의 클러스터 기반 메일시스템들이 갖는 문제점인 부하관리와 결합허용 문제 그리고 구조의 복잡성 등을 해결하고 웹 환경과 연결하여 사용할 수 있는 클러스터 기반의 웹-메일 시스템을 설계한다. 그리고 클러스터 기반의 메일 시스템의 특수한 경우인 전체 메일 발송에 관한 효율적이 구조도 제안하였다.

이 논문의 구성은 2장에서 클러스터 기반의 메일 시스템에 관한 관련연구를, 3장에서는 새로운 시스템 구조를 설계하고 이 구조에 대한 동작 시나리오를 보인다. 4장에서는 시스템에 대한 결합허용과 부하관리를, 5장에는 전체메일을 효율적으로 전송하기 위한 방법을 살펴보고 6장에서 결론 내린다.

2. 관련 연구

2.1 EarthLink

NFS를 이용한 분산메일 시스템으로 SMTP서버 그룹과 POP 서버그룹, 파일서버그룹, 인증서버로 나뉘어서 수행된다. SMTP와 POP서버에 접근한 사용자는 우선 인증서버에서 인증과정을 수행한 후 파일서버를 NFS를 이용하여 메일을 가져오거나 보내는 작업을 수행한다.

EarthLink의 단점은 인증서버의 부하증가와 서버고장에 따

2.2 Porcupine

EarthLink의 단점을 보완하여 모든 노드가 동등한 작업을 수행하면서 사용자의 메일박스를 여러 노드로 분리하여 보관함으로써 부하균형을 이루었고, 인증 작업도 한 노드가 담당하는 것이 아니라 전체 모든 노드가 동시에 인증작업을 수행할 수 있다. Porcupine은 hard state와 soft state의 두 가지 형태의 자료구조와 user, mailbox, user profile, replication manager 등 총4개의 manager를 두었다[2].

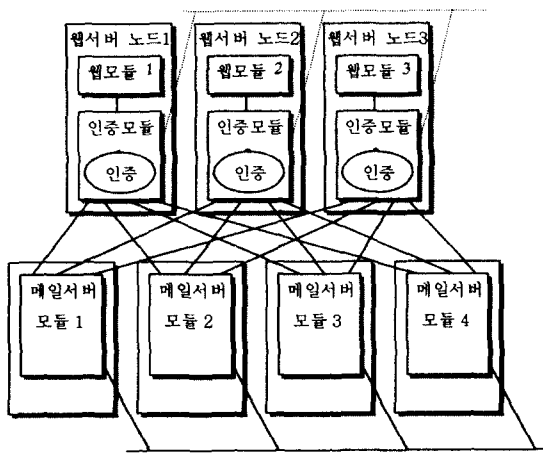
Porcupine의 단점은 지나치게 부하균형을 관리하여 구조가 복잡하다.

3. 메일 시스템의 구조

설계한 클러스터 기반의 웹 메일 시스템은 앞에서 2장에서 살펴본 Earthlink와 Porcupine을 섞어놓은 형태로 <그림 1>과 같이 2계층의 노드와 3계층의 모듈로 구성되어 있다.

웹 모듈은 웹 부분과 프로그래밍 언어로 코딩된 인증모듈 부분과의 연결고리 역할을 하며 웹 프로그램 자체라고도 볼 수 있다. 인증모듈은 웹 부분과 메일서버 사이를 연결하는 부분으로 인증과, 부하 관리를 함께 맡는 부분이다. 메일서버모듈은 메일서버 노드에 위치하고, 인증모듈과 웹 모듈과는 다른 노드에 위치하고 사용자 정보 관리와 메일박스 관리를 한다.

인증모듈(웹서버)과 메일서버모듈(메일서버)사이에는 TCP/IP로 항상 연결되어 있고 메일서버모듈(메일서버) 사이에도 정보를 주고받을 수 있게 TCP/IP로 연결 되어있다. 즉 TCP 연결이 끊어지는 노드는 더 이상 존재하지 않은 노드가 된다. 인증모듈과 인증모듈 사이에는 UDP 프로토콜로 이루어진 링 방식으로 구성되어 있다.



< 그림 1 > 웹-메일시스템의 전체 구조

3.1 자료구조

자료구조는 Porcupine과 비슷한 구조로 사용자 요약정보와 사용자 등록정보로 구성된다. 여기서 주 노드란 고객의 메일이 저장되는 노드(메일박스)로 사용자 등록정보를 가지고 있는 노드를 말하고, 부 노드란 노드의 파손을 대비하여 사용자 등록정보와 메일박스를 복사한 노드를 말한다.

- 사용자 요약정보 램 : 인증을 위한 자료구조로 사용자 요약정보 DB중 자주 사용하는 고객을 램에 저장한 형식으로 사용자 ID, 비밀번호, 주 노드번호, 부 노드 번호로 구성되고 각 인증모듈마다 같은 값을 갖지는 않는다.
- 사용자 요약정보 DB : 사용자 요약정보 램과 같은 구조로 웹 메일 시스템의 모든 고객을 가지고 있으며 웹 메일 시스템에 있는 모든 인증모듈의 사용자 요약정보 DB의 내용을 동일하다.
- 사용자 등록정보 : 주 노드에 위치하며 사용자 요약정보 내용을 포함하고 주민등록번호, 주소, 전화번호 등 고객에 관한 상세 정보를 가지고 있고 고객이 사용하는 메일들을 관리하기 위한 변수가 저장된다.
- 메일요약정보 파일 : 고객이 현재 가지고 있는 메일들에 관한 요약정보(총 메일 개수, 각 메일파일 이름, 메일 번호, 메일 제목, 메일 상태, 송신자 주소, 수신 시각, 크기)를 저장하고 있는 파일이다.
- 메일요약정보RAM : 메일 요약정보 파일 중 많이 사용하는 고객을 RAM에 올린 것이다.

3.2 동작 시나리오

메일 시스템이 동작하는 방식으로 웹을 이용한 사용자 중심의 시나리오이고 인증, 등록, 삭제, 변경 등이 있다.

3.2.1 인증 시나리오

웹 서버에서 사용자 번호와 비밀번호를 입력하면 DNS의 라운드 로빈 방식에 의하여 한 웹서버의 인증모듈에 사용자 번호와 비밀번호를 전달한다. 이 인증모듈에서는 사용자 번호와 비밀번호를 비교하여 연결 세션을 만들고 해당 사용자의 주 노드의 메일서버모듈고도 연결을 한다. 그리고 웹 서버에 인증이 처리되었음을 알린다.

3.2.2 메일 읽기 시나리오

웹을 통하여 메일을 읽는 과정이다. 우선 인증과정을 수행하게 된다. 그러면 인증과정 중 라운드로빈 방식으로 선택된 웹서버의 인증모듈과 사용자의 주 노드의 메일서버 모듈간의 연결이 이루어지고 이 연결을 통해 현재 메일박스의 내용(메일 이름, 발송자, 제목)이 웹 브라우저로 전송이 되고 사용자는 이중 원하는 메일을 선택하면 웹서버를 통해 웹 브라우저의 화면에 메일을 보여준다.

3.2.3 메일 도착, 메일 발송 시나리오

메일의 도착은 웹서버를 거치지 않고 DNS에서 바로 메일서버의 sendmail이 메일을 받아 내부사용자 여부를 검사한 후 메일박스에 저장한다.

메일의 발송은 웹을 통해 작성된 메일을 sendmail을 통해 발송한다.

3.2.4 등록 시나리오

고객이 웹 브라우저의 가입 버튼을 누르면 웹 서버는 라운드 로빈 방식으로 한 인증서버에 고객등록 서식을 비밀번호 없이 보낸다. 고객등록 서식은 고객등록을 위해 특수하게 미리 정해진 번호(일반 사용자는 사용할 수 없는)로 로그인 하게 된다. 이후에 인증서버는 부하 관리자를 호출하여 부하가 적은 메일서버(주 노드)를 선택하여 세션을 설정한다. 그 후 고객은 웹서버에서 정보를 입력하고 이것은 메일 서버의 사용자 등록정보에 저장한다. 메일서버는 부 노드에 사용자 등록정보에 보내고 요약정보를 모든 인증서버에 보낸다.

3.2.5 삭제, 변경 시나리오

등록 시나리오와 같이 삭제, 변경 시 고유의 번호로 인증을 받고 등록과 같은 시나리오로 동작한다.

4. 부하관리와 결합허용

부하균형과 결합허용측면에서 EarthLink, Porcupine과 새로 설계한 구조에 대해 알아본다.

4.1 부하관리

부하정보는 인증모듈에서 보관하고 있으며 새로운 고객을 등록 할 경우와 일부 메일 서버 노드에 집중되는 부하를 발견했을 때 메일을 옮기는 작업을 하기 위해 사용된다.

부하정보를 얻는 방법은 두 가지가 있다. 하나는 메일서버가 인증모듈과 접속 시 부하정보를 붙여서 보내는 방법이고 하나는 인증 모듈간에 링 구조로 부하 정보를 교환하여 최신 정보로 갱신하여 사용한다. 이 시스템에서는 두 가지를 모두 사용하여 부하 정보를 관리한다.

인증서버모듈 안에 부하를 관리하는 부하관리자가 있고 여기서 부하에 관한 정보를 메일서버모듈로부터 수집하여 관리한다. 인증서버모듈 간의 동기화는 링 구조로 메일서버에 대한 부하정보를 타임스탬프, 토큰과 함께 담아서 보내게 된다.

Porcupine은 각각 고객의 메일박스를 여러 노드에 분할하여 지나치게 복잡한 구조로 부하균형을 시도한 반면 이 방법은 메일서버모듈과 인증모듈사이에 통신을 할 때 간단한 부하정보를 인증모듈의 부하관리자에게 넘기는 방법으로 부하를 갱신하고 인증모듈 간에 동기화도 수행하므로 정확한 정보를 간단한 구조로 설계하였다.

4.2 결합허용

메일서버의 하위계층의 고장과 웹서버의 상위계층의 고장에 대처방법이 구분되어 있다.

4.2.1 메일서버의 결합허용

웹 서버와 메일서버사이와 메일서버사이에는 TCP로 연결이 되어있기 때문에 메일서버가 고장을 일으키면 모든 웹 서버와 메일서버는 고장 난 노드를 자체적으로 제거할 수 있다. 인증모듈에서는 현재 토큰을 소유하고 있는 인증모듈이 임시로 결합허용을 위한 관리자가 된다. 관리자가 하는 일은 다음과 같다.

- 관리자가 가진 부하정보를 이용하여 고장 난 노드와 관련이 있는 모든 사용자의 자료구조를 변경시킨다.
- 또 관리자는 새로운 자료구조를 관리자라는 표시와 함께 모든 노드(인증모듈+ 메일서버모듈)에 브로드캐스트 한다.
- 각 메일서버모듈은 사용자등록정보를 수정한 후 수정된 자료구조에 맞게 고객의 메일박스를 이동시킨다.
- 각 인증서버 모듈은 새로운 자료구조를 받는다. 하지만 즉시 적용시키지는 않고 정상적으로 받았다는 결과보고만 관리자에게 한다.
- 메일서버모듈에서 정상적으로 고장 난 노드를 제거하고 모든 메일박스를 처리했을 경우 관리자에게 보고를 한다.
- 관리자가 모든 노드에게 정상적으로 자료구조 수정이 되었다는 결과를 받으면 정상적으로 갱신작업을 종료했다는 것을 알리기 위해 전 노드에게 브로드캐스트를 한다.
- 관리자는 부하관리를 위한 토큰을 다시 전송한다.

4.2.2 웹 서버의 결합 허용

웹 서버의 고장은 처리가 간단하다. 메일서버모듈 방향의 연결은 다른 인증서버모듈이 처리해 주면 되기 때문에 신경을 쓸 필요가 없고 인증서버 모듈간의 연결은 부하균형을 위한 token을 부하정보와 함께 전송하므로 쉽게 발견할 수 있다.

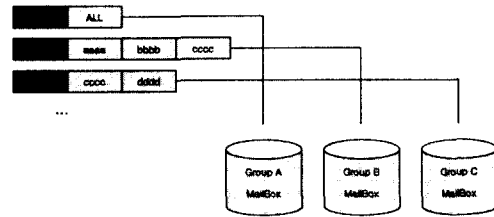
Porcupine의 경우 각각 사용자의 메일박스가 전 노드에 골고루 있기 때문에 한 노드의 고장은 거의 모든 고객에게 영향을 미쳐 User map[2]의 변경이 심하다. 메일을 복사했을 경우 메일의 이동이 복잡하지만, 이 시스템은 영향을 받는 고객의 수가 적고, 상위 구조(인증서버모듈)쪽에서 관리자역할을 수행해 작업이 간단해 진다. 하지만 복사를 하지 않았을 경우 porcupine은 많은 사용자가 일부의 메일만 분실하지만, 이 시스템은 일부 고객의 메일을 전부 분실할 위험이 있다.

5. 브로드캐스팅

여기서 말하는 브로드캐스팅이란 메일 시스템의 운영자가 공지사항 같이 모든(일부) 고객에게 전체메일을 보내는 것을 말한다. 모든 고객에게 특별한 구조 없이 메일을 직접 보낸다면 가입자의수 만큼의 메일을 직접 보내기 때문에 속도와 공간측면에서 많은 낭비가 생긴다.

이를 해결하기 위한 방법으로 공통의 메일박스를 두고 관리하는 방법이 있다. 공용 메일박스를 두고 고객은 항상 메일을 읽을 때 공용메일박스를 검색하여 메일을 읽는 방식이다. 이 방법을 유지하기 위해서는 이 브로드캐스팅된 메일을 특정사용자가 읽었는지 확인하기 위한 자료구조가 필요하다. 이 자료구조는 메일 당 하나의 자료구조를 두는 방법과 사용자등록정보에 두는 방법이 있다. 그러나 사용자가 많아질 경우 메일마다 하나의 자료구조를 갖는 경우는 현실성이 없다. 또 공용메일의 삭제는 가입만 하고 사용하지 않는 고객이 많기 때문에 일정시간이 지났을 때 삭제하는 방법 외에는 계속 메일을 갖고 있어

야 한다.



< 그림 2 > 그룹화 한 브로드캐스팅 방법

<그림 2>과 같이 전자잡지나 구독 신문 같은 전체 메일이 아닌 일부고객에게만 보내는 경우에는 그룹을 만들어서 해당 사용자가 해당 그룹에 가입하고 그룹마다 하나씩의 공용 메일박스를 갖는 형태로 사용한다. 이 때 전체 사용자도 하나의 그룹으로 보고 전체 공지사항은 전체 그룹에 전송한다.

6. 결론

현재 진행중이거나 만들어진 분산메일 시스템인 EarthLink와 Porcupine을 간단한 구조를 알아보았고 Earthlink의 단점인 불안한 부하균형, 인증서버의 무결성, Porcupine의 단점인 지나친 부하균형으로 발생하는 복잡한 구조를 없애고 이 분산 메일 시스템을 웹 환경과 합친 웹 메일 시스템에 적용시켰다.

웹과 인증부분을 상위단계의 클러스터링으로 묶고 메일시스템 부분을 하위 계층으로 나눠서 고장이 발생하였을 경우 주도가 되어 고장을 처리하는 노드를 확실하게 결정할 수 있고 구조도 간단하게 되었다. 부하의 균형문제의 경우 Porcupine이 한 고객의 메일박스를 여러 개로 분할하면서까지 미세하게 부하를 맞춤으로서 구조가 복잡해지는 문제를 메일박스를 한군데로 고정하고 부하의 불균형이 일어날 경우 사용자를 다른 노드로 옮기는 작업을 하여 문제를 해결하였다.

다른 문제로 사용자를 여러 그룹으로 나누어서 메일박스를 둬서 공지메일이나 잡지구독 같은 단체 메일을 보낼 때의 공간이나 수행능력의 저하를 막았다.

참고문헌

- [1] Nick Christenson, Tim Bosserman, and David Beckemeyer, "A Highly Scalable Electronic Mail Service Using Open Systems", In Symposium on Internet Technologies and Systems. USENIX 1997
- [2] Yasushi Saito, Brian N. Bershad, and Henry M. Levy, "Manageability, Availability, and Performance in Porcupine: A Highly Scalable, Cluster-Based Mail Service", ACM Transactions on Computer Systems, Vol. 18, No. 3, pp. 298-332, August 2000
- [3] David Wood, "programming Internet Email", O'REILLY & Associates, August, Inc., 1992
- [4] Brrrrell, A. D., Hisgen, A., Jrtisn, C., Mann, T., and Swart, G., "The Echo distributed File Systems", In 15th Symposium on Operating System Principle. ACM, 1995
- [5] Brisco, T. P. "DNS support for load balancing", RFC1794: <http://www.cis.ohio-state.edu/htbin/rfc/rfc1794.html>, 1995