SDSR : Ad-hoc 망에서의 안정성을 제공하는 Dynamic Source Routing

김혜원° 박용진
한양대학교 전자전기컴퓨터공학부
{hwkim°, park}@hyuee.hanyang.ac.kr

Stable Dynamic Source Routing in Ad-hoc network

Hye-Won Kim°, Yong-Jin Park
Division of Electrical and Computer Engineering, Hanyang University

요 약

기존에 제시된 ad hoc 라우팅 프로토콜에는 안정성에 대한 부분이 고려되어 있지 않다. 본 논문에서는 기존의 DSR ad hoc 라우팅 프로토콜에 안정성을 접목한 SDSR 라우팅 프로토콜을 제시한다. SDSR은 DSR에 안정성 제공을 위해 abnormal node detector와 neighbor table이라는 것을 추가한다. abnormal node detector는 네트워크 내에 abnormal 노드를 탐지해 네트워크에서 고립시켜 네트워크에 안정성을 제공하고 neighbor table에 있는 priority를 값에 따라 이웃 노드에서 들어온 패킷을 처리함으로써 효율적인 처리 능력을 제공한다. 본 논문에서는 abnormal node detector와 neighbor table을 통해 어떤 방식으로 네트워크에 안정성을 제공하는지 살펴보도록 하겠다.

1.서 론

Ad hoc 네트워크는 고정된 라우터나 호스트, 무선 기지국을 가지지 않는 무선 노드들만 구성된 망이다. 이동 노드들이 노드들 사이의 통신을 위해 고정된 기지국을 가지고 있지 않기 때문에 각 노드가 라우터로 동작하여 패킷을 노드에서 다른 노드로 전송하는 시스템으로 구성되어 있다. 이런 ad hoc 망에서는 노드들이 자유롭게 이동 가능하게 때문에 경로 설정과 유지에 어려움이 존재한다. 이런 ad hoc 환경에 적절한 라우팅 프로토콜은 중요한 역할을 한다. 현재 ad hoc 망에서의 라우팅을 위해 제안되고 있는 라우팅 알고리즘은 크게 두 가지가 있다. 각 노드에 routing table을 유지해 수시로 네트워크의 변화를 수정하는 table-driven 방식이 있고 다른 노드로 패킷 전송이 필요할 때 경로를 설정하고 유지하는 on-demand 방식이 있다. table-driven 방식에는 DSDV가 있고 on-demand 방식에는 AODV와 DSR이 있다. 이러한 라우팅 프로토콜은 안전한 네트워크 환경에서 동작할 수 있게 만들어진 라우팅 프로토콜이다. 안정성에대한 고려나 네트워크에 참여한 노드에 대한 신뢰도를 고려하지 않았다. 만일 네트워크 내에 오동작 하는 노드들이 존재한다면 DoS 공격이나 잘못된 패킷 유발, 잘못된 경로로 패킷 전송과 같은 형태의 문제들이 발생할 수 있다. 본 논문에서는 이러한 문제에 대응할 수 있는 안정된 ad hoc 라우팅 프로토콜을 제시하고자 한다. 현재 IETF에서 표준화로 고려되고 있는 DSR에 안정성을 접목한 SDSR을 제안한다. section 2에는 간단하게 기존의 DSR에 대하여 설명하고 section 3에서는 SDSR에 대해 알아보도록 한다.

2.관련 연구

2.1 Dynamic Source Routing (DSR)

DSR은 on-demand로 작동하는 소스 라우팅 프로토콜이다. 이 프로토콜에는 경로 발견과정(route discovery)과 경로 유지과정(route maintenance) 두 가지 과정으로 구성되어 있다[1].

```
[when a node N receives a packet]
if(RREQ packet)
  if(packet' RREQ ID != RREQ ID in node cache)
    if(addresses in RREQ' route record != node' address)
      if(RREQ' destination address != node' address)
        a node attachs to node' address in route record
        and broadcast the network
      else
        send RREP
    else
      discard packet
  else
    discard packet
else if(RREP packet)
  if(node N is the source node)
    select route
  else
```

```
    forward a packet to the source node
else if(ERROR packet)
  if(node N is the source node)
    if the source node needs the route, initiate
    the route discovery
  else
    remove error node' address in route cache
    and forward a packet to the source node
else /* if data or other control packet */
  process the packet using the underlying
  routing protocol
```

그림1. DSR 경로 발견과 유지 과정에 대한 동작 코드

3. 제안하는 SDSR

3.1 가정

가. 노드 사이의 모든 링크는 양방향 통신을 한다.

나. 각 노드는 promiscuous mode를 지원한다.

3.2 abnormal 노드 정의

ad hoc 네트워크는 라우팅과 포워딩을 하기 위해 모든 이용 가능한 노드를 사용하여 전체적인 네트워크의 처리량을 최대화 한다. 라우팅에 관련된 노드가 많아질 수 록 사용할 수 있는 대역폭 증가, 짧은 라우팅 경로와 네트워크 분열 가능성을 줄일 수 있다. 그러나 이러한 노드가 비정상적 동작을 한다면 네트워크 내에 악영향을 미치게 된다.

4가지로 간단하게 오동작하는 노드를 분류하면 아래와 같다.

-과부하가 걸린 노드: 패킷 전달을 위한 대역폭, cpu cycle, 저장 공간의 부족으로 오작동을 유발하는 노드

-selfish 노드: 자신과 관련되지 않은 패킷 전송으로 인해 자신의 자원을 낭비하게 되는 것을 꺼려하는 노드

-malicious 노드: 불필요한 패킷 발생 혹은 버림으로 인해 DoS 공격을 유발하는 노드

-고장난 노드: 패킷 전송과 관련해 노드 내부에 하드웨어 혹은 소프트웨어에 결함을 가지고 있는 노드

위에서 노드에 과부하가 걸린 경우와 고장난 노드의 경우 네트워크에서 쉽게 일어날 수 있는데다 의도적으로 네트워크에 자원을 낭비하지 않는다. 이 경우에는 단순히 경로 유지과정을 이용해 해결할 수 있다. 그러나 selfish 노드나 malicious 노드는 의도적으로 네트워크 자원을 낭비하고 악영향을 미치게 된다. 이 두 가지 형태의 성격을 띤 즉 네트워크 내에 불필요한 패킷을 발생하면서 전송 능력이 떨어지는 노드를 abnormal 노드라고 정의하겠다.

3.3 abnormal node Detector와 neighbor table

본 논문에서는 주어진 time_interval내에서 불필요하게 잦은 발생으로 생성된 패킷을 garbage 패킷이라고 정의한다. garbage packet은 abnormal node detector에 의해서 탐지되며 처리된다. abnormal node detector에는 각 패킷의 type과 특성에 따라 다른 기준값을 가지며 기준값에 따라 이웃노드에서 garbage 패킷 발생여부를 파악할 수 있다. 본 논문에서 제시된 abnormal node detector의 설명에서는 DSR 내에 쉽게 문제를 유발할 수 있는 브로케스트 성격을 띤 RREQ 패킷을 위주로 설명하도록 하겠다. RREQ 패킷은 ad hoc 네트워크에서 소스 노드가 목적지 노드까지 경로를 찾기 위해 브로트캐스팅 되는 패킷이다. DSR에서는 불필요한 RREQ의 발생을 막기 위해 백-오프 알고리즘을 사용한다. 예를 들어 중간 노드의 이탈로 인해 RREQ를 발생한 노드가 정해진 시간 내에 reply를 수신하지 못할 경우 지수 함수 백-오프 알고리즘을 사용하여 충분한 시간 간격을 가지고 RREQ를 재전송한다. 만약 이러한 알고리즘을 무시하고 abnormal 노드가 네트워크 내에 존재해 불필요하게 RREQ를 발생하게 되면 네트워크 자원을 낭비하게 된다. 이러한 문제점을 해결하기 위해 abnormal node detector를 사용한다. abnormal node detector의 역할은 위에서 언급한 abnormal 노드를 탐지해 보다 효율적인 처리 능력과 안정된 네트워크를 유지할 수 있도록 해준다.

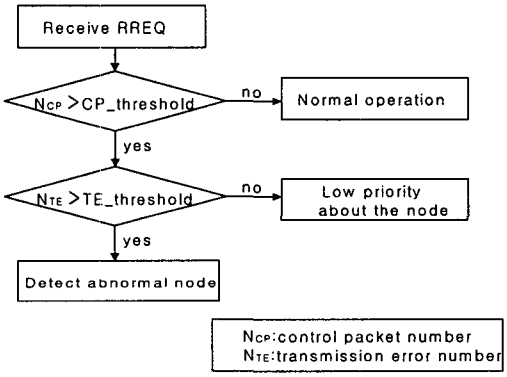

그림 2. abnormal node detector의 동작 모델

가. abnormal node detector 동작 방법

특정한 time_interval 내에 어떤 목적지를 향해 소스노드에서 발생한 RREQ의 발생횟수가 CP_threshold를 넘었다면 해당 노드가 abnormal 노드인지 의심하게 된다. 그래서 해당 노드에 대한 neighbor table 안에 있는 transmission error를 참조하여 전송률을 판단한다.

첫째, transmission error가 TE_threshold를 넘지 않았다면 해당 노드에 대한 priority를 낮추게 된다. neighbor table 에는 모든 이웃 노드들에 대한 priority값을 가지고 있는데 정상적으로 동작하는 노드는 양의 값을 가지지만 abnormal 노드로 판단되는 노드는 음의 값을 가지게 된다. 각 노드는 priority값에 따라 패킷 처리를 한다. 그럼으로써 정상적으로 동작하는 노드에게는 높은 priority를 주어 패킷 처리에 대한 더 많은 기회를 제공하고 abnormal 노드는 아니지만 잦은 RREQ의 발생으로 네트워크의 자원을 불필요하게 소비하는 노드에 대해서는 낮은 priority를 주어 적은 패킷 처리 기회를 제공하게 된다. 결과적으로 특정 노드의 불필요한 패킷 발생으로 인한 네트워크 내의 병목현상을 막을 수 있다.

둘째, transmission error가 TE_threshold를 넘어서게 되면 해당 노드는 abnormal 노드로 판단한다. abnormal 노드를 발

건한 노드는 이웃 노드들에게 notify message를 사용해서 abnormal 노드의 존재를 알리게 된다. notify message를 받은 이웃 노드들은 abnormal 노드라고 판단되는 노드의 control packet number와 transmission number 값을 확인한다. 만약 각각의 해당 값이 CP_threshold±α, TE_threshold±β 값을 넘었다면 abnormal 노드로 판단하고 priority 값을 음수 값을 준다. abnormal 노드로 판정된 노드에서 발생된 패킷은 이웃 노드들에 의해서 폐기된다. abnormal 노드를 경로로 가지는 source 노드에게는 route error message를 전송하여 새로운 경로를 찾는다. 이와 같은 방법으로 네트워크 내에서 abnormal 노드를 고립 시키게 된다. 이러한 abnormal node detector 기능은 모든 노드에 존재하며 이와 함께 neighbor table이라는 것도 같이 동작한다[3][4].

나. neighbor table의 구성 요소

radio channel의 브로드캐스트 특성 때문에 전송할 때 마다 해당 패킷의 목적지가 아니더라도 이웃들의 packet을 감청 할 수 있다. 해당 노드는 promiscuous 모드를 통해 이웃들을 지나가는 모든 패킷을 감청해 주위에 어떤 이웃 노드들이 존재하는지 알 수 있다. 감청한 패킷에서 이웃노드들에 주소를 얻을 수 있고 이 정보를 neighbor table에 기록한다. neighbor table에는 IP address, control packet number, transmission error, priority 필드를 가지고 있다. 여기서 control packet number는 이웃 노드들이 발생하는 control 패킷의 수이다. 여기서 priority란 control packet의 발생 횟수와 비례해 매겨진 값이다.

IP address	Control packet num	Transmission error	priority
166.104.29.51	1	2	9
166.104.29.50	10	15	-10

그림3. neighbor table

```
[Detector operates after a node N stores and transfers a packets]
if (node is overhear when a node N' neighbor transfer a packet)
    if(a packet in buffer == a overheard packet)
     /*sucess transmission */
       delete a packet in buffer
    else (a packet in buffer for the time interval>
          time_threshold)     /*fail to transmission*/
        discard a packet
        update the value in the transmission error field
```

그림 4. 에러 탐지 동작 코드

neighbor table안에 있는 transmission error 값은 promiscuous mode를 사용하여 전송에러를 탐지 하여 transmission error 값을 변경한다. 그림5에서 A노드는 B노드를 통해 C노드로 패킷을 전송할 때 A노드는 전송한 패킷을 버퍼에 저장해 둔다. B노드가 C노드를 향해 패킷을 전달할 때 해당 패킷을 감청한 후 저장되어 있는 패킷과 같은지 여부를 확인한다. 만약 같다면 패킷이 올바르게 전송된 것이므로 버퍼 안에 있는 패킷을 삭제한다. 버퍼 안에 패킷이 정해진 시간을 초과해서 남겨져 있을 경우 패킷 전송에 담당하는 B노드에 대해 transmission error 값을 높여 준다[2].

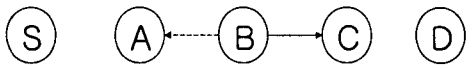

그림5. promiscuous mode의 예

다. 해결되는 문제점

앞에서 제시된 abnormal node Detector와 neighbor table을 통해서 아래와 같은 문제점들을 해결할 수 있다.

-Denial of service

DoS 공격 결과로 abnormal 노드가 네트워크의 대역폭을 차지하게 된다. abnormal 노드는 불필요하게 경로 요구 패킷을 자주 생성해 다른 노드가 네트워크의 자원을 사용하지 못하게 한다.

- Energy consummation

에너지는 ad-hoc 네트워크에서 중요한 파라메타이다. 배터리 충전식 장비들은 필요할 때만 패킷 전송을 시도해 에너지를 절약하려고 한다. abnormal 노드는 불필요한 패킷 발생은 다른 노드로의 전송을 통해 배터리를 소모하는 시키게 된다.

4.결론

본 논문에서 기존의 ad hoc 라우팅 프로토콜과는 다른 안정성을 고려한 라우팅 프로토콜을 제시하고 있다. 각 노드에는 neighbor table와 abnormal node detector를 가지고 있다. abnormal node detector는 neighbor table 안에 있는 전송 에러, control 패킷 발생률 값을 이용해 이웃 노드에 abnormal 노드가 존재하는지 여부를 파악한다. 만약 자신의 이웃 노드들 중에 abnormal 노드가 존재한다고 판정되면 이웃 노드들에게 이 사실을 통보해 abnormal 노드를 네트워크에서 고립시킨다. neighbor table에는 abnormal node detector에 의해 매겨진 priority 값에 따라 이웃 노드에서 들어온 패킷을 처리한다. 논문에서 제시된 SDSR 라우팅 프로토콜은 네트워크 내에 abnormal 노드가 존재할 때 유용한 메커니즘이다.

5.참고자료

[1]C.E. Perkins, "ad hoc networking", Addison-wesley Publishing company, 2001

[2]Sergio Marti, T.J.Giuli, Kevin Lai, and Mary Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Network." In proceedings of MOBICOM2000

[3]Panagiotis Papadimitratos, Zygmunt J. Haas, Prince Samar "draft-papadimitratos-secure-routing-protocol-00.txt"

[4]Sonali Bhargava, Dharma P. Agrawal "Security Enhancements in AODV protocol for wireless Ad Hoc Networks"