

SIP 환경에서의 Registration Event package 구현

조현규⁰ 정영면 장춘서
금오공과대학교 컴퓨터공학과
{blackjo⁰ webtain1}@cespcl.kumoh.ac.kr, csjang@kumoh.ac.kr

Implementation of Registration Event package in SIP Environment

Hyun Gyu Jo⁰ Young Myun Jung Choon Seo Jang
Dept. of Computer Engineering, Kumoh National Institute of Technology

요 약

SIP(Session Initiation Protocol) 기반의 VoIP(Voice over IP) 시스템의 구성 요소에서 레지스트라(Registrar)는 사용자들의 Registration 정보들을 저장 및 관리하는 역할을 한다. 네트워크상에서 이 Registration 상태는 관리자의 정책 등에 따라 동적으로 변화하게 되므로 사용자는 변경되는 상태 정보를 알 필요성이 생긴다. Registration Event Package는 UA(User Agent)가 Subscriber로서 Registration 상태의 변경에 대한 통지를 요청할 수 있고 알림(Notification)을 통해 얻을 수 있는 매커니즘을 제공한다. 본 논문에서는 SIP 환경에서 이러한 Registration Event Package를 구현하여 사용자가 동적으로 변화되는 Registration 상태 정보를 알림을 통해 받을 수 있도록 하였다.

1. 서론

SIP 기반의 VoIP 시스템에서 사용자들간에 통신의 세션 설립과 유지에 필요한 기능 가운데 Registration은 사용자가 통신에 사용할 실제 연결 주소와 같은 정보 등을 REGISTER 메소드를 통하여 네트워크 상의 레지스트라(Registrar)에 등록, 수정 및 삭제하는 것을 의미한다. UA(User Agent)는 통신을 위한 하나 이상의 Contact 주소를 가질 수 있으며 Registration시 이 Contact 주소들은 "address-of-record"인 SIP URI와 바인딩 된다[1].

네트워크 상에서 Registration 상태는 관리자의 정책 등과 같은 요인에 따라 동적으로 변화될 수 있다. 예로써 관리자가 보안의 필요상 인증을 갱신하기 위해 Registration의 유효 시간을 변경시키거나 사용자가 주어진 시간내에 갱신이 없을 경우 더 이상 사용할 수 없는 상태가 될 수도 있다. 따라서 사용자는 자신의 Registration 상태 변화에 대하여 알 필요성이 생긴다.

본 논문에서는 SIP 환경에서 이러한 동적으로 변화하는 사용자의 Registration 상태를 알 수 있는 기능을 제공하는 Registration Event Package를 구현

하였다.

본 논문의 구성은 다음과 같다. 2장에서 Registration Event Package 관하여 설명하고 3장에서는 구현된 시스템의 전체적인 내용을 다루며 4장에서 결론을 맺는다.

2. Registration Event Package

Registration Event Package는 UA가 Subscriber로서 등록(Subscription)과 알림(Notification)을 통해 변화되는 Registration의 상태 정보를 얻을 수 있는 매커니즘을 제공한다[2].

SIP 기반에서는 등록과 알림을 위해 확장된 메소드인 SUBSCRIBE와 NOTIFY가 사용되며 SUBSCRIBE 요청 메시지의 Event 헤더값은 "reg"를 가진다. 그림 1은 일반 사용자가 MIME 타입을 "application/reginfo+xml"로 하여 자신의 Registration 상태에 대하여 등록하는 SUBSCRIBE 요청 메시지의 예이다. 만약 메시지내에 Expires 헤더가 생략될 시에 이를 받은 레지스트라는 3600초를 기본값으로 처리한다.

```
SUBSCRIBE sip:johg@sip2.kumoh.ac.kr SIP/2.0
Via: SIP/2.0/UDP 202.31.137.176;branch=z9hG4bk813876976
Max-Forwards: 70
To: sip:johg@sip2.kumoh.ac.kr
From: sip:johg@sip2.kumoh.ac.kr;tag=2089007545
Call-ID: 2073439175@sip2.kumoh.ac.kr
Event: reg
Accept: application/reginfo+xml
Cseq: 2 SUBSCRIBE
Contact: sip:johg@sip2.kumoh.ac.kr
Content-Length: 0
```

그림 1. SUBSCRIBE 요청메시지의 예

레지스트라는 다이제스트 인증[3,4]을 통과한 SUBSCRIBE 요청에 대해 Registration State Machine(RSM)과 Contact State Machine(CSM)의 변화에 따라 알림이 발생하는 시점을 결정한다.

그림 2에 RSM의 상태 전이도를, 그림 3에 CSM의 상태 전이도를 각각 보였다.

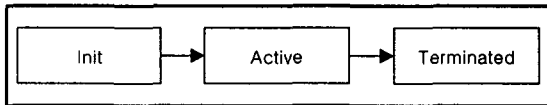


그림 2. Registration State Machine(RSM)

RSM의 상태 변화 및 알림의 발생은 다음과 같다. 먼저 address-of-record에 대해 Contact 주소가 처음 등록될 때 Init 상태에서 Active 상태로 이동하며 이때 NOTIFY 요청 메시지를 전송한다. 이 Active 상태 동안에 동일한 사용자로부터 REGISTER 요청이 여러차례 들어와도 계속 이 상태에 있으므로 이 경우에는 알림이 발생하지 않는다. 만약 Registration된 모든 Contact 주소가 expire 되면 RSM은 Terminated 상태로 이동하고 변경 상태를 담은 NOTIFY 요청 메시지를 전송한다.

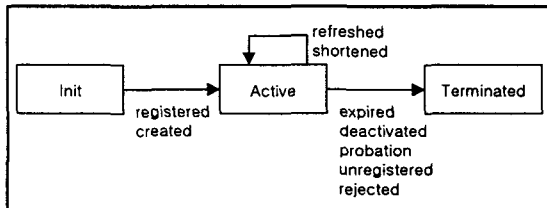


그림 3. Contact State Machine(CSM)

CSM에서 상태의 변화를 일으키는 이벤트는 아래의 경우에 발생하여 Subscriber에게 NOTIFY 메시지를 전송한다.

- registered : 새로운 Contact 주소를 가진 REGISTER 요청 메시지가 들어올 경우

- refreshed : 동일한 Contact 주소를 가진 REGISTER 요청 메시지가 들어올 경우
- shortened : 관리자가 해당 Contact 주소에 대해 expiration 시간을 변경할 경우
- expired : 해당 Contact 주소에 대해 유효 시간내에 갱신이 없을 경우
- unregistered : expiration 시간이 0 인 REGISTER 요청 메시지가 들어올 경우
- deactivated : 관리자가 Contact 주소를 삭제하지만 사용자가 이 주소를 다시 Registration 할 수 있는 경우
- rejected : 관리자가 Contact 주소를 삭제하고 사용자는 이 주소를 다시 Registration 할 수 없는 경우
- probation : 관리자가 Contact 주소를 삭제하지만 사용자가 이 주소를 일정 시간의 경과 후 다시 Registration 할 수 있는 경우

RSM과 CSM의 변화에 따른 reg NOTIFY 요청 메시지의 전송시 변경된 Registration 상태의 정보는 메시지의 바디에 "application/reginfo+xml" 포맷으로 내장되며 그림 4는 사용되는 예이다.

```
<?xml version='1.0'?>
<reginfo xmlns='urn:ietf:params:xml:ns:reginfo'
  version='0' state='full'>
  <registration aor='sip:johg@sip2.kumoh.ac.kr'
    id='1544629105' state='terminated'>
    <contact id='1' state='active' event='registered'
      duration-registered='3600'
      q='0.5'>sip:johg@irt1.kumoh.ac.kr</contact>
    <contact id='2' state='terminated' event='expired'
      duration-registered='3600'
      q='1.0'>sip:johg@irt2.kumoh.ac.kr</contact>
  </registration>
</reginfo>
```

그림 4. Registration 상태 정보

그림 4에서 reginfo 요소의 속성인 version은 0부터 시작하여 새로운 NOTIFY 메시지가 보내질 때마다 1씩 증가한다. state 는 full 또는 partial의 값을 갖는다. registration 요소의 속성인 aor은 address-of-record를 가르키며 id는 registration 요소들 사이에서 유일한 값을 가지고 state는 RSM의 상태중 하나의 값을 나타낸다. CSM에 대응되는 contact 요소의 속성은 Contact를 구분하는 id, CSM의 상태를 나타내는 state 및 상태의 변화를 일으킨 event가 기본적으로 포함되고 옵션인 duration-registered는 Contact가 address-of-record

와 바운딩되는 시간을 의미하며 q 값은 Registration 된 다른 Contact들과 비교되는 상대적인 우선 순위 값을 가진다. 또한 shortend 이벤트 발생시에 필요한 expires 속성과 probation 이벤트를 위한 retry-after 속성이 존재한다.

3. 시스템 구현

시스템의 구성은 SIP 기반의 레지스트라에 Registration 정보에 대한 등록을 수신하고 알림을 발생하는 기능을 추가하여 프록시 서버와 함께 linux OS 상에서 구현하였다. 또한 UA를 확장하여 Subscriber로서 Registration Event Package에 등록 요청이 가능하고 알림을 받는 기능을 수행하도록 구현하였다. UA는 자바를 사용하여 GUI 형태로 작성하였으며 PC 윈도우즈 상에서 구현하였다.

레지스트라는 UA의 Registration과 Subscription 요청의 처리를 위해 Registration State Machine(RSM)과 Contact State Machine(CSM)을 포함시켜 구현하였다.

이를 위한 테이블은 “Registration State Table(RST)”, “Contact State Table(CST)”, “Reg Subscription Table(RegST)” 로 구성하였다.

테이블 RST는 RSM이 관리하는 테이블로서 REGISTER 요청 메시지 내의 address-of-record에 해당하는 SIP URI, 식별자 및 RSM의 변화되는 상태 정보를 저장한다. 이때 하나의 RSM은 하나의 address-of-record와 대응된다. 각 REGISTER 요청의 각 Contact에 대응되는 테이블 CST는 CSM에 의해 관리되며 식별자와 Contact 정보, 변화되는 CSM의 상태 및 이를 변화시킨 이벤트 등을 저장한다. 테이블 RegST는 UA가 Registration Event Package에 등록시 등록 대상 마다 생성 또는 갱신되며 Subscriber의 정보와 버전 값 및 서비스의 유효시간 등을 저장한다.

그림 5는 구현된 시스템에서 UA가 레지스트라에 Registration하기 위해 전송하는 REGISTER 요청 메시지가 인증을 거치는 실제 화면이다. 시스템에서 사용한 모든 인증은 RFC 2617 문서에 기반한 다이제스트 인증을 사용하였다.

인증을 통과한 메시지는 파싱되어 각 테이블에 저장되고 이에 맞추어 RSM과 CSM의 상태는 변화하게 된다.

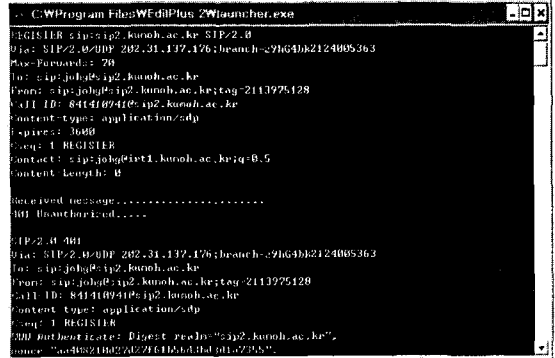


그림 5. REGISTER 요청 메시지

그림 6은 사용자 “johg”가 자신의 Registration 상태를 알기 위해 Registration Event Package에 등록하는 과정에서 다이제스트 인증 정보를 포함하여 SUBSCRIBE 요청 메시지를 재 전송하는 실제 화면이다.

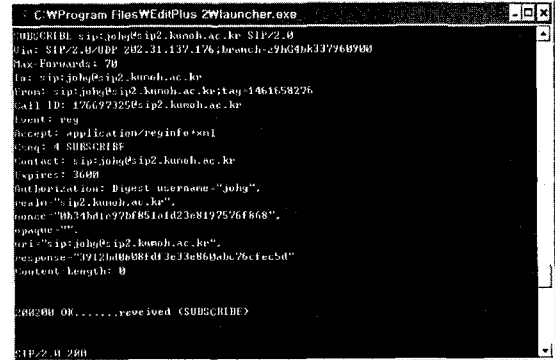


그림 6. 인증정보를 포함한 SUBSCRIBE 메시지

레지스트라는 SUBSCRIBE 요청 메시지를 파싱하여 테이블 RegST에 정보를 저장 또는 갱신하고 RSM과 CSM의 변화에 따라 알림의 발생 시점을 결정한다.

만약 알림이 발생하면 Registration 정보를 “application/reginfo+xml” 포맷으로 처리하여 NOTIFY 요청 메시지의 바디에 내장시키고 Subscriber에게 전송한다. 그림 7은 구현된 시스템에서 사용자가 기존에 Registration한 Contact와 동일한 “sip:johg@irt2.kumoh.ac.kr”의 주소로 다시 Registration할 경우 CSM에서 refreshed 이벤트가 발생하여 이에 해당하는 NOTIFY 요청 메시지를 전송하는 실제 화면이다. 그림 7에서 사용자 “johg”는 두개의 Contact 주소를 가지고 있으며 id가 1인

“sip:johg@irt1.kumoh.ac.kr”은 Registration이 된 후 현재 active 상태로 남아 있음을 알 수 있다.

```

[연락 2002.11.13.03:33]
Content-Disposition: ...
NOTIFY : sip:johg@irt1.kumoh.ac.kr SIP/2.0
Via: SIP/2.0/UDP 202.31.130.33;branch=29164082659651376
To: From: sip:johg@irt1.kumoh.ac.kr;tag=45224711
Cc: sip:johg@irt1.kumoh.ac.kr;tag=1422524711
Call-ID: 12495657499-1ip2.kumoh.ac.kr
Content-Type: application/reginfo+xml
Seq: 6 NOTIFY
Content-Length: 529

<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo"
  version="5" state="partial">
  <registration src="sip:johg@irt1.kumoh.ac.kr" id="313139419" state="active">
    <contact id="1" state="active" event="registered"
      duration-registered="3600"
      q="0.5"> sip:johg@irt1.kumoh.ac.kr</contact>
    <contact id="2" state="active" event="refreshed"
      duration-registered="3600"
      q="1.0"> sip:johg@irt2.kumoh.ac.kr</contact>
  </registration>
</reginfo>
message received.....
SIP/2.0 200
Via: SIP/2.0/UDP 202.31.130.33;branch=29164082659651376
To: From: sip:johg@irt1.kumoh.ac.kr;tag=1422524711
Cc: sip:johg@irt1.kumoh.ac.kr;tag=45224711
Call-ID: 12495657499-1ip2.kumoh.ac.kr
Content-Type: application/reginfo+xml
Seq: 6 NOTIFY
    
```

그림 7. NOTIFY 요청 메시지

Subscriber는 알림을 받아 처리하기 위해서 Registration의 address-of-record에 해당하는 테이블을 구성한다. 각 테이블은 NOTIFY 메시지 바디의 reginfo의 하위 요소인 registration의 id 로 구분하였다. 테이블의 내용은 registration 요소의 하위 요소인 contact 요소를 가진다.

Subscriber는 최초 수신한 NOTIFY 요청 메시지 내의 reginfo 요소의 버전 값이 0 이므로 이 값으로 초기화하고 이후 NOTIFY 메시지를 수신하면 그림 8과 같이 버전 및 테이블을 관리한다.

4. 결론

본 논문에서는 SIP 환경에서의 구성 요소의 하나인 레지스트라에 대한 Registration Event Package를 구현함으로써 UA가 Registration 뿐만 아니라 Subscriber로서 Registration Event Package에 등록을 할 수 있고 변화되는 Registration 상태 정보를 알림을 통해 받을 수 있도록 하였다.

시스템 내에는 RSM과 CSM을 구현하여 발생하는 이벤트를 감시하고 또한 상태의 변화를 관리하며 알림의 발생 시점을 결정할 수 있도록 하였다.

알림이 발생할 때마다 NOTIFY 요청 메시지의 바디에 포함되는 Registration의 상태 정보는 “application/reginfo+xml”을 타입으로 하는 XML 포맷으로 처리하였다.

참고문헌

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, “Session Initiation Protocol”, RFC 3261, June 2002.
 [2] J. Rosenberg, “A Session Initiation Protocol (SIP) Event Package for Registrations”, draft-ietf-sipping-reg-event, October 28, 2002.
 [3] Jonathan Rosenberg, “SIP security Mechanisms Update”, dynamicsoft presentation, 05, 2002.
 [4] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, L. Stewart, “HTTP Authentication: Basic and Digest Access Authentication”, RFC 2617, June 1999.

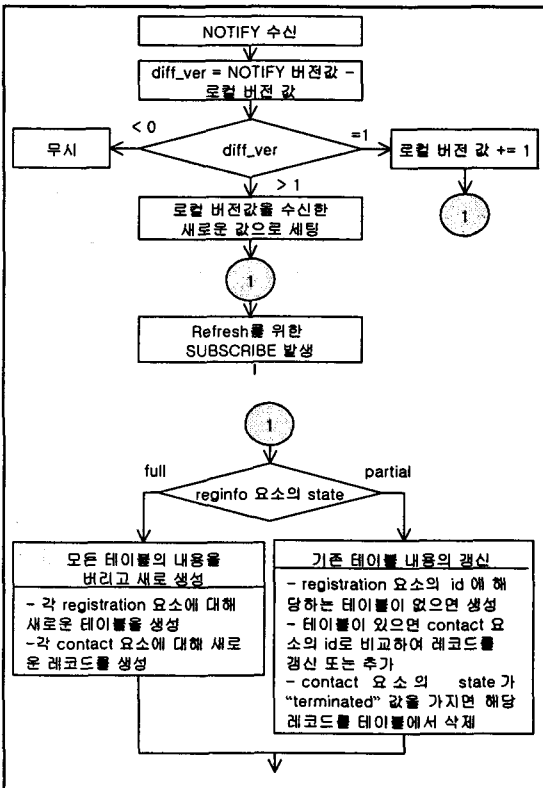


그림 8. Subscriber의 로컬 버전 및 테이블 관리