

# POP3와 HTTP를 이용한 메일 전송 에이전트

유경종, 이부권  
경상대학교 컴퓨터과학과

## A Mail Transport Agent Using POP3 and HTTP

Kyeong-Jong Yoo, Bu-Kwon Lee  
Dept. of Computer Science, Gyeongsang National Univ.  
E-mail: mac30@rtp.gsnu.ac.kr, bklee@gsnu.ac.kr

### 요약

현재 인터넷 서비스 중에서 가장 많이 사용되는 서비스는 이메일이다. 본 연구에서는 POP3로부터 전송받은 이메일 메시지와 HTTP를 통해 접속한 웹메일 시스템의 이메일 메시지를 분석하여 제목, 보낸이, 본문 메시지와 특정 첨부파일을 분리해낸 후 USB에 연결된 아바타 로봇시스템에 전송 및 재생하는 시스템을 설계 및 구현하였다. 텍스트 메시지는 음성 변환 모듈을 통해 음성 파일로 변환하여 전송하게 되고, 특정 확장자의 첨부 파일은 바로 전송된다.

### 1. 서론

현재의 인터넷 서비스 중에서 많이 사용되는 것은 WWW과 이메일이다. 특히 이 중에서 이메일은 전세계적으로 가장 널리 사용되는 인터넷 서비스인데, 단순한 메시지 전송을 넘어서 파일 전송, 메일링 리스트 등 다양한 분야에서 사용되고 있다[1].

초창기 이메일 서비스는 영어권 국가에서만 사용되었기 때문에 인코딩할 필요가 없었다. 하지만 비영어권 국가의 언어를 전송하기 위해서는 Base64나 Quoted Printable 등과 같은 인코딩 방법을 사용하여 아스키 텍스트로 변환하여 메일을 전송해야 한다. 그리고 여러 형태의 메시지를 한 메일 메시지에 전송하기 위해서 MIME(Multipurpose Internet Mail Extentions)을 이용하여 메일을 전송한다[2].

현재 메일을 확인하기 위해서는 POP3, IMAP 등의 프로토콜이 필요한데, 대부분의 웹메일 시스템의 경우 이러한 서비스를 제공하지 않는 경우가 많다.

이 경우에는 HTTP를 통해 이들 웹메일 시스템에 접속, 인증 절차를 거친 후 HTTP를 통해 온 메시지를 분석하여 메일 메시지를 확인할 수 있다.

본 연구에서는 POP3로 전송된 메일 메시지와 HTTP로 웹메일 시스템에 접속하여 분석한 메일 메시지를 분석하여 메일 내용을 음성으로 변환, 아바타 로봇에 전송하여 메일을 확인하고 음성 인식 기능을 통해 메일을 삭제할 수 있는 시스템을 제안한다.

### 2. 관련연구

#### 2.1 메일 관련 프로토콜

전자 메일을 송수신할 때 사용되는 프로토콜은 여러가지가 있으나 보통 POP3, IMAP, SMTP 등을 사용하여 메일을 확인하고 전송할 수 있다.

POP3(Post Office Protocol – Version 3)는 메일 서버에서 사용자의 PC로 메일을 다운로드 받아 메일을

확인할 때 사용되는 프로토콜이다[3].

IMAP(Internet Message Access Protocol)는 POP3와 기본적으로는 유사하지만 POP3가 메일을 클라이언트에 저장하는 것과는 달리 서버에 메일을 저장하여 모든 메일을 다운로드 받지 않고 필요한 메일만 읽을 수 있으며, 삭제도 가능하다[4].

SMTP(Simple Mail Transfer Protocol)은 메일 서버 간에 메일을 전송할 때 사용하는데, 사용자가 메일을 보낼 경우 이 프로토콜을 통해 메일을 주고 받을 수 있다[5].

## 2.2 메일 메시지와 인코딩

메일 메시지는 대부분의 메일 클라이언트 프로그램을 이용하여 전송할 때 기본적으로는 HTML 형식의 텍스트 메시지로 전송되고, 일반 텍스트와 HTML 형식을 택일 또는 두 개의 형식 모두 같이 보낼 수도 있다. 첨부 파일이 있는 경우나 메일 클라이언트에서 제공되는 기본 그림을 포함하는 경우 등 복합적인 메시지를 한꺼번에 전송하는 경우에는 MIME를 이용하여 메일 메시지 구성하여 전송할 수 있다.

초기 메일 클라이언트 프로그램에서는 영어권 국가에서 사용되는 문자만을 사용하였기 때문에 7비트 아스키만으로 충분했지만, 현재는 여러 국가에서 메일 서비스를 사용하기 때문에 아스키 문자로 메일을 전송하는 경우에 메일이 깨질 수 있다. 한글과 같은 비영어권 문자는 8비트로 표현되어야 하기 때문에 8비트를 아스키 코드로 표현하기 위해 여러 가지 인코딩 방법이 사용된다[6]. 유닉스에서는 이진 파일 첨부를 위해 uuencode, uudecode를 사용하였고, 현재 가장 널리 사용되는 인코딩 방법은 Base64 방식과 Quoted Printable이다. 그래서 한글이 포함된 경우 제목과 메일 본문 내용은 반드시 디코딩 되어야 한다.

웹메일 시스템에서 메일을 확인하는 경우에는 표준화된 프로토콜이 없기 때문에 HTTP로 전송된 메시지에서 헤더를 제거하고[7] 본문의 내용인 HTML을 분석하여 필요한 부분을 분리해내어야 한다.

## 3. 메일 전송 에이전트

이 절에서는 메일 전송 에이전트 전체 구성과 설계, 그리고 POP3와 HTTP를 통해 전송된 메시지를 분석하는 방법에 대해 알아본다.

### 3.1 메일 전송 에이전트

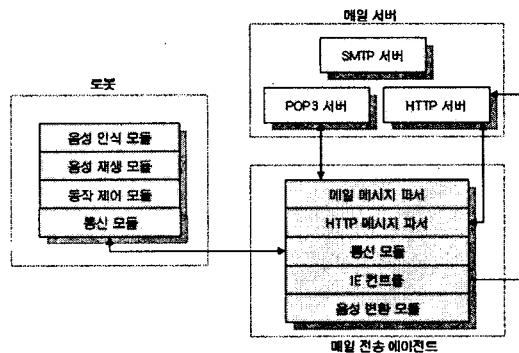


그림 1. 시스템 전체 구성도

그림 1에서 POP3 서버를 통해 받아온 메일 메시지는 메일 메시지 파서를 통해 제목과 메일 내용을 분리해낸다. 분리된 메일 내용은 음성 변환 모듈을 통해 음성 파일로 변환되고 아바타 로봇에 전송되어 재생할 수 있다. 메일 삭제시에는 아바타 로봇에게 삭제 명령을 내리면 음성 인식 모듈에 의해 명령어 코드로 바뀌어 다시 메일 전송 에이전트에 전송, POP3 삭제 명령을 통해 해당 메시지를 삭제한다.

HTTP의 경우에는 먼저 IE 컨트롤(CHTMLView)에 띄워진 로그인 화면을 통해 사용자 인증을 한 다음, HTTP 메시지 파서를 통해 설정된 서버의 메일리스트가 표시되는 페이지로 접속하여 제목에 링크된 URL에 다시 접속한다. 이후 그에 해당하는 메시지를 분리해내어 음성으로 변환 및 전송, 재생할 수 있다.

### 3.2 메일 및 HTML 파싱

#### ① POP3 메일 메시지 파싱

POP3로 전송받은 메일 메시지를 파싱하기 위해 서 CPop3Socket, CPop3MailParser 클래스를 구현하였다.

CPop3Socket 클래스에는 POP3 서버에 접속하여 메일 메시지 개수를 알아내고, 그 개수만큼 메시지를 가져오는 기능이 포함되어 있다. CPop3MailParser 클래스는 가져온 메시지에서 보낸 사람 이름, 제목과 본문을 분리해내어 만약 Base64와 같은 인코딩 기법으로 인코딩된 경우 디코딩하는 부분까지 포함되어 있다. 요즘의 메일 클라이언트의 경우 기본 메시지가 HTML 형식으로 되어있는 경우가 많기 때문에 HTML 태그 삭제 기능도 구현되어 있다. 그리고 본 시스템에서 아바타 로봇에 전송되는 내용은 보낸이와 제목, 메일 본문, 그리고 .aro 확장자를 가진 첨부 파일만 필요하다. .aro 확장자 외의 확장자를 가진 파일은 별도로 저장되지만 아바타 로봇에 전송되지 않는다.

두 클래스는 다음과 같은 멤버 함수들을 가진다.

```
BOOL Connect(CString HostName, CString User, CString Password);
BOOL Disconnect();
BOOL Statistics(int& NumberOfMails, int& TotalMailSize);
BOOL Delete(int Msg);
BOOL GetMessageSize(int Msg, DWORD& Size);
BOOL Retrieve(int Msg, CPop3MailParser& Message);
```

```
CString GetMessageText()
CString GetHeader() const;
LPCSTR GetBody() const;
CString GetSubject()
CString GetFrom()
CString GetDate()
CString GetReplyTo() const;
CString GetContentType() const;
CString GetBoundary() const;
LPCSTR GetAttachement();
CString GetFileName();
CString Base64Decoder(const CString& Content);
CString RemoveHTML(CString& Content);
```

## ② HTTP 데이터 파싱

현재 대부분의 웹메일 시스템의 경우 POP3나 IMAP을 통해서 메일 확인을 하는 것을 지원하지 않는다. 이 때문에 웹메일 시스템의 메일을 확인하기 위해서는 인증 절차를 거친 다음 HTTP로 접속하여 HTML 메시지 안에 포함된 메일 제목과 메일 내용을 분리해내어야 한다.

거의 대부분의 웹메일 시스템은 메일 제목과 내용들은 table 태그 내의 tr과 td 태그에 들어있으므로 몇 번째 table 태그가 실제 메일과 관련된 내용인지를 본 시스템에서 설정해주어야 한다. 이 정보는 웹메일 시스템 모두 다르기 때문에 새로운 웹메일 서버가 추가될 때마다 설정해 줄 필요가 있다. 그리고 메일의 리스트에서 출력되는 메일 개수, 메일의 리스트, 메일의 본문, 메일이 삭제되는 URL도 설정해주어야 한다. 설정과 관련된 내용은 클래스의 생성자를 통해 초기화 시켜 줄 수 있다.

메일 내용의 경우는 메일 리스트의 제목 링크의 href 인수값을 따라 한번 더 HTTP 접속을 한 후에 얻을 수 있다. 메일 내용이 HTML 형식으로 되어 있는 경우에는 태그를 제거한 후에 음성 변환을 해야 한다.

첨부 파일은 POP3로 받아온 메시지와 마찬가지로 모든 파일을 다 전송받지만 .aro 확장자의 파일만 아바타 로봇에 전송하게 된다.

다음은 CHtpParser의 멤버 함수들이다.

```
BOOL Connect(CString HostName, CString URL);
CString GetSubject();
CString GetName();
CString GetURL();
CString GetContent();
LPCSTR GetAttachement();
CString GetFileName();
int GetMailCount();
CString RemoveHTML(CString& Content);
BOOL Delete(int Msg);
```

다.

#### 4. 구현 및 결과

본 연구에서 사용된 시스템은 Windows XP 환경에서 Visual C++ 6.0, WinSock2 API를 이용하여 구현하였다.

인터넷 메일 시스템은 기존의 메시지 전달 뿐만 아니라 파일 전송, 토론 그룹, 사용자 인증 등의 다양한 용도로 사용될 수 있는데, 본 메일 전송 에이전트로 아바타 로봇에 음성을 전달하여 메일 확인 및 삭제하는 방법을 제안하였다. 그림 2는 본 시스템의 메인 화면이다. 실행을 시키면 POP3의 경우 설정된 서버에 접속하여 아바타 로봇에서 메일이 차례대로 재생된다. 웹메일의 경우는 인증 과정 후 차례대로 메일 내용을 확인할 수 있다.

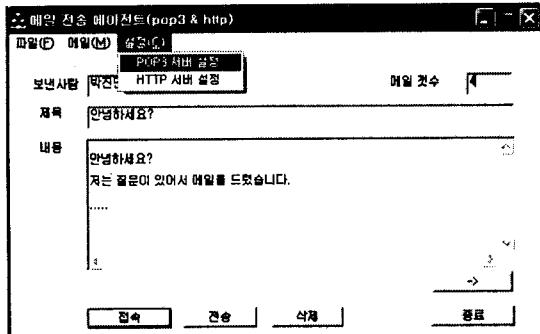


그림 2. 메일 전송 에이전트 메인 화면

단점으로는 POP3와 IMAP을 지원하지 않는 웹메일 시스템에서 화면 인터페이스가 바뀌는 경우 파싱 방법을 수정해주어야 한다는 것이다.

#### 5. 결론 및 향후 과제

본 연구에서는 POP3로 전송받은 메일 메시지와 HTTP로 전송받은 웹메일 메시지를 파싱을 하는 기법을 제안하였고, 이를 음성 변환 및 전송하여 아바타 로봇에 전달하여 재생할 수 있도록 하였다.

향후 IMAP를 통해서도 메일을 수신할 수 있도록 하고, 다른 여러 웹메일 서버까지 지원할 계획이

#### [참고문헌]

- [1] David Wood, "Programming Internet Email", O'Reilly, 1999.
- [2] RFC 1341, MIME(Multipurpose Internet Mail Extentions), <http://www.faqs.org/rfcs/rfc1341.html>, June 1992.
- [3] RFC 1939, Post Office Protocol – Version 3, <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1939.html>, May 1996.
- [4] RFC 2060, Internet Message Access Protocol – Version 4recv1, <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2060.html>, December 1996.
- [5] RFC 821, SMTP(Simple Mail Transfer Protocol), <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc0821.html>, August 1982.
- [6] Craig Hunt, "TCP/IP 네트워크 관리", O'Reilly, 1998.
- [7] RFC 2068, Hypertext Transfer Protocol – HTTP/1.1, <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2068.html>, January 1998.