DETEKSI E-MAIL PALSU DENGAN MEMPERGUNAKAN *HEADER* E-MAIL

M. Didik R. Wahyudi,
Teknik Informatika
Institut Sains & Teknologi AKPRIND Yogyakarta

Abstrak

E-mail is a medium of communication to send and receive messages. As one of the media that is most commonly used to exchange information, e-mail is vulnerable to security threats. One of the security threat to e-mail is the aspect of the authenticity of e-mail received. Habits of users who receive e-mail read directly and trust where it came from the e-mail and any contents of the e-mail. E-mail that was sent using certain technique, can manipulate the original e-mail. Recipient of e-mail will assume that e-mail came from someone who is, as there is in the header from. This can cause misunderstanding, so that will hurt many parties that did not do so.

This study will elaborate the characteristics and the characteristics of e-mail based on false information in the e-mail header. After a false e-mail is identified, then this e-mail will get special treatment, such as marked or even removed immediately.

Identification done with a header decipher e-mail that contains all information about the e-mail that is sent further validated with a variety of criteria, false e-mail that has been specified above.

Kata kunci : False E-mail, E-mail Filtering, E-mail Security

Intisari

E-mail merupakan media komunikasi untuk mengirim dan menerima pesan. Sebagai salah satu media yang paling sering dipakai untuk bertukar informasi, e-mail sangat rentan terhadap ancaman keamanan. Salah satu ancaman keamanan e-mail adalah dari aspek keaslian e-mail yang diterima. Kebiasaan user yang menerima e-mail adalah langsung membaca dan mempercayai dari mana datangnya e-mail tersebut dan apapun isi e-mail tersebut. Pengiriman e-mail yang dilakukan dengan teknik tertentu, dapat memanipulasi asal e-mail. Penerima e-mail akan mengira bahwa e-mail tersebut berasal dari seseorang yang sebagaimana terdapat pada header *from*. Hal ini dapat menimbulkan salah paham, sehingga akan merugikan banyak pihak yang sebenarnya tidak melakukannya.

Penelitian ini akan melakukan penelusuran terhadap ciri-ciri dan karakteristik *e-mail* palsu berdasarkan informasi yang terdapat pada header e-mail. Setelah e-mail palsu ini diidentifikasi, maka *e-mail* ini akan mendapat perlakuan khusus seperti ditandai atau bahkan langsung dihapus.

Identifikasi dilakukan dengan menguraikan header e-mail yang berisi semua informasi mengenai e-mail yang dikirimkan selanjutnya divalidasi dengan berbagai kriteria e-mail palsu yang sudah ditentukan diatas.

Kata kunci : E-mail Palsu, Filtering E-mail, Keamanan E-mail

PENDAHULUAN

Komunikasi selalu dilakukan antar individu satu dengan yang lain. Pada era internet, komunikasi mulai berubah diantaranya chatting, teleconference hingga e-mail. Belakangan ini, pemakaian *Electronic-Mail* atau yang sering disebut e-mail semakin popular. E-mail dapat diartikan sebagai media komunikasi untuk pengiriman dan penerimaan pesan yang tersimpan dalam komputer dalam bentuk ASCII[4]. Komunikasi yang dilakukan dengan e-mail relatif lebih efisien daripada komunikasi dengan surat-menyurat secara konvensional. Proses pengiriman dan penerimaan relatif cepat dan akurat. Berbagai brosur yang berisi tawaran suatu produk dapat dikirimkan dalam waktu singkat kepada sejumlah e-mail yang menjadi anggota suatu klub (mailing list). Pada awalnya e-mail memiliki format text, namun pada saat ini e-mail juga berisi kode-kode HTML, gambar dan suara dengan dukungan MIME[6], sehingga suatu halaman web dapat dikirimkan melalui e-mail. Hal ini akan membuat peran e-mail akan menjadi lebih dari sekedar media komunikasi.

Sistem email terdiri dari dua komponen utama, yaitu *Mail User Agent* (MUA), dan *Mail Tranfer Agent* (MTA). MUA merupakan komponen yang digunakan oleh pengguna email. Biasanya dia yang disebut program mail. Contoh MUA adalah Eudora, Netscape, Outlook, Pegasus, Thunderbird, pine, mutt, elm, mail, dan masih banyak lainnya lagi. MUA digunakan untuk menuliskan email seperti

halnya mesin ketik digunakan untuk menulis surat jaman dahulu.

MTA merupakan program yang sesungguhnya mengantar email. Biasanya dia dikenal dengan istilah mailer. MTA ini biasanya bukan urusan pengguna, akan tetapi merupakan urusan dari administrator. Contoh MTA antara lain postfix, qmail, sendmail, exchange, MDaemon, Mercury, dan seterusnya.

Pada dasarnya e-mail memiliki dua model yaitu e-mail tanpa *attachment* dan e-mail dengan *attachment*. Suatu e-mail dibuka dan dikirimkan dari *client* dengan mengambil yang tersimpan dalam server. Dalam e-mail *client*, tidak semua atribut e-mail ditampilkan semua, hanya beberapa atribut penting yang ditampilkan, misalnya *From*, *To, Date, Subject* dan *body message* [7]. Berikut ini contoh e-mail :

```
From: “Odix Wahyudi”
<odix@localhost.localdomain>
To: <dita@localhost.localdomain>
Subject: Apa Kabar
Date: Wed, 9 Oct 2002 00:44:08
+0700 (WIT)
Message-ID:
<Pine.LNX.4.33.0210090043220.966
8-100000@localhost.localdomain>
----------------------
Hallo, ini coba kirim e-mail
Iya.. coba e-mail
```

E-mail di atas dapat diterjemahkan sebagai : e-mail datang dari **Odix Wahyudi** dengan alamat e-mail odix@**localhost.localdomain**. Dikirimkan ke dita dengan alamat e-mail **dita@localhost.localdomain**. Subject e-mail : **Apa Kabar** dengan tanggal e-mail **Wed, 9 Oct 2002 00:44:08 +0700 (WIT)**. Kode pesan **Pine.LNX.4.33.0210090043220.9668-100000@localhost.localdomain**. Isi pesan adalah **Hallo ini coba kirim e-mail; Iya.. coba e-mail**. Pada e-mail di atas, *header* tidak ditampilkan. Berikut ini format e-mail dengan *header* :

```
Return-Path:
<odix@localhost.localdomain>
Received: from localhost
(odix@localhost)
by localhost.localdomain
(8.11.6/8.11.6) with ESMTP id
g98Hi9W09673
for
<dita@localhost.localdomain>;
Wed, 9 Oct 2002 00:44:09 +0700
Date: Wed, 9 Oct 2002 00:44:08
+0700 (WIT)
From: “Odix Wahyudi”
<odix@localhost.localdomain>
To: <dita@localhost.localdomain>
Subject: Apa Kabar
Message-ID:
<Pine.LNX.4.33.0210090043220.966
8-100000@localhost.localdomain>
MIME-Version: 1.0
Content-Type: TEXT/PLAIN;
charset=US-ASCII
----------------------
Hallo, ilustrasi mengirim e-mail
```

Komunikasi dengan e-mail mempunyai beberapa aturan baku yang harus dipenuhi agar e-mail tersebut dapat dibaca dalam berbagai *platform*. Aturan baku tentang format e-mail terdapat pada RFC822 dan RFC2822. Aturan ini meliputi model e-mail yang disusun dari *fields* e-mail dengan teknik tertentu.

Selain untuk mengirim pesan sebagaimana surat konvensional, e-mail juga biasa dipergunakan untuk mengirim suatu *file* dengan mempergunakan fasilitas *attachment*. File yang dikirim dilampirkan dalam e-mail yang dikirim. Jika pada e-mail tanpa *attachment, body message* hanya berisi satu *body message*, maka pada e-mail jenis ini selain berisi pesan utama juga berisi *attachment* yang dicantumkan dalam bentuk deretan karakter yang sudah di-*encode* dalam bentuk ASCII dengan menggunakan **7BIT, 8BIT, BINARY, BASE64, QUOTED-PRINTABLE** dan beberapa coding lainnya [9].

Sistem e-mail sudah sangat pentingnya sehingga banyak orang akan mengeluh jika sistem e-mail tidak dapat bekerja. Bahkan banyak bisnis yang dilakukan dengan menggunakan e-mail. Dapat dibayangkan jika sistem e-mail tidak dapat bekerja dalam waktu yang lama [8]. Ada beberapa masalah keamanan yang terkait dengan sistem email, yaitu :

- Disadap
- Dipalsukan
- Disusupi (virus)
- Spamming
- Mailbomb
- Mail relay

Dari beberapa permasalahan keamanan e-mail diatas, yang akan dibahas dalam penelitian ini adalah permasalahan keamanan e-mail yang berhubungan dengan e-mail yang dipalsukan. Berikut ini langkah-langkah yang akan ditempuh dalam penelitian ini :

Langkah 1 : Ekstraksi header e-mail dengan mempergunakan *script* PHP

Langkah 2 : Cara membuat e-mail palsu

Langkah 3 : Identifikasi e-mail palsu dengan header

Langkah 4 : Penyaringan e-mail palsu

PEMBAHASAN

Proses ekstraksi header e-mail mempergunakan fungsi IMAP yang terdapat pada *script* PHP. IMAP (Internet Message Acces Protocol) merupakan suatu protokol yang dipergunakan untuk mengakses e-mail. POP (Post Office Protocol) juga protocol untuk manipulasi e-mail. Pemakaian fungsi IMAP untuk ekstraksi header e-mail karena fungsi ini memberikan kemudahan dalam ekstraksi field-field e-mail. POP dapat juga dipergunakan dengan memasukkan server POP pada fungsi IMAP.

Proses ekstraksi header e-mail dengan mempergunakan fungsi IMAP dari *script* PHP untuk e-mail yang tidak memiliki *attachment* untuk memperoleh struktur header e-mail sebagaimana dari dokumen RFC822 adalah sebagai berikut :

```
$mailbox =
imap_open("{odx.myserv.com}INBO
X","gaza","gaza99");
$header = imap_header($mailbox,
2);
$message = imap_body($mailbox,
2);
$strmsg =
@imap_fetchstructure($mailbox,
1);
imap_close($mailbox);

//menampilkan header
$from_array =
imap_mime_header_decode($header
->fromaddress);
$to_array =
imap_mime_header_decode($header
->toaddress);
print("From : "
.htmlspecialchars($from_array[0
]->text). "<br>\n");
print("To : "
.htmlspecialchars($to_array[0]-
>text). "<br>\n");
print("Date : " . $header->date
. "<br>\n");
print("<br>Message-ID : "
.htmlspecialchars($header-
>message_id). "<br>\n");
print("<br>Content-Type : "
.$strmsg->type. "/" .$strmsg-
>subtype. "<br>\n");
print("Subject : " . $header-
>subject . "<p>\n");

//message
print("<p>Isi Pesan :");
$msgarray = explode("\r\n",
$message);
for($i=0;$i<count($msgarray);$i
++)
print("<br>$msgarray[$i]");
```

Script ini berfungsi untuk menguraikan *field* e-mail. Variabel $mailbox berisi informasi kotak surat (*mailbox*) dari parameter *account* yang dituju. Fungsi imap_open membaca *mailbox* sasaran. Parameter {odx.myserv.com}INBOX menunjukkan nama *server* mail sedangkan parameter odx adalah *account* e-mail yang akan dilihat dan gaza99 password *account* e-mail yang dimaksud. Variabel $header mengambil *header* e-mail dari variabel $mailbox, dengan nomor pesan 2. Variabel $message berisi isi pesan yang diambil dari variabel $mailbox, dengan nomor pesan 2.

Field alamat pengirim diambil dengan menggunakan fungsi imap_mime_header_decode yang menunjuk pengirim header→fromaddress yang disimpan dalam variabel *array* $from_array, dan kemudian diambil detail *array* (alamat pengirim) dengan $from_array[0]→text. Untuk alamat penerima masih mempergunakan fungsi yang sama, hanya saja penunjuknya berbeda yaitu header→toaddress yang disimpan dalam *array* $to_array, dan kemudian alamat tujuan diambil dengan $to_array[0]→text. Untuk tanggal pengiriman mempergunakan fungsi header→date dan subyek (judul) pesan, diambil dengan fungsi header→subject. Isi e-mail diambil dari variabel $message. Untuk mendapatkan format yang sama dengan pengiriman, maka variabel $message diuraikan kembali dengan mempergunakan fungsi explode untuk mengambil per baris. Tiap baris dipisahkan oleh karakter \n\r. *Script* di atas jika dijalankan akan menghasilkan keluaran :

```
From : Odix Wahyudi
<odix@odx.myserv.com>
To : <gaza@odx.myserv.com>
Date : Wed, 5 Feb 2003 11:21:03
+0700 (WIT)
Message-ID:
<Pine.LNX.4.33.0302051120100.255
2-100000@odx.myserv. com>
Content-Type:TEXT/PLAIN
Subject : Tanpa Attachement
---------------------------Isi
Pesan :
Ilustrasi mengirim e-mail tanpa
attachment
```

Cara membuat e-mail palsu

Cara membuat e-mail palsu tidak terlalu sukar. Tuliskan informasi yang salah di header dari email. Misalnya konfigurasikan sistem email dengan mengatakan bahwa pengirim adalah si-doel@hotmail.com. Email yang palsu ini kemudian kita serahkan kepada MTA untuk dikirimkan ke tempat yang dituju. Maka MTA akan melakukan perintah tersebut. Misalnya dengan membuat sebuah file "email-palsu.txt" dengan isi sebagai berikut :

```
To: siapasaja@dimanasaja.com
From: si-doel@hotmail.com
Subject: email palsu
Saya akan coba kirim email
palsu. Perhatikan
header dari email ini.
```

Langkah selanjutnya setelah file ditulis, panggil MTA (contoh sendmail) untuk mengirimkan email ke alamat "user01@training". Email akan dikirimkan ke "user01@training", tanpa memperdulikan isi field "To:" yang ada dalam berkas tersebut. Berikut ini *script* yang dipergunakan untuk mengirimkan e-mail palsu tersebut :

```
/usr/sbin/sendmail
user01@training     <     email-
palsu.txt
```

Selain dengan cara tersebut, ada cara lain yang bisa dipergunakan untuk mengirimkan e-mail, yaitu dengan langsung berbicara ke MTA yang dituju dengan menggunakan protokol SMTP.

```
Unix% telnet mailserver 25
HELO localhost
MAIL FROM: saya@hotmail.com
RCPT TO: user01
DATA
354 Enter mail, end with "." on a line by itself
To: haha@hotmail.com
From: hoho@hotmail.com
Subject: palsu

nih palsu
.

250 HAA20290 Message accepted for delivery
QUIT
```

Gambar 1 : Mengirim e-mail palsu dengan Telnet SMTP

Identifikasi e-mail palsu dengan header

E-mail palsu yang sudah dikirimkan, akan diterima seperti e-mail lain yang asli. Sekilas e-mail palsu dan asli ini tidak bisa dibedakan. Gambar 1 dan gambar 2 memberikan contoh e-mail palsu dan e-mail asli :

```
From y3dips@plasa.com Wed Nov 19 18:13:01 2003
From: y3dips@plasa.com  Add to Address Book
Subject: ini email aslinya
To: talent_spidey@yahoo.com
Date: Thu, 20 Nov 2003 09:13:01 +0700



ini email asli nih


SALAM
====
y3dips
```

Gambar 2 : E-mail yang asli

From y3dips@plasa.com Wed Nov 19 18:31:14 2003
From: y3dips@plasa.com Add to Address Book
Subject: ini email palsunya
To: talent_spidey@yahoo.com
Date: 20 Nov 2003 02:31:14 -0000

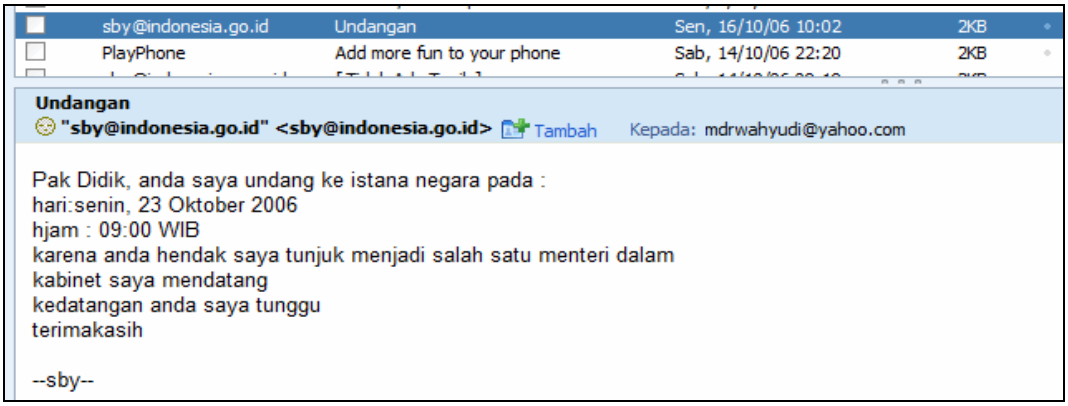
ini palsu !! :P

SALAM
=====
y3dips

Gambar 3 : E-mail yang palsu

Untuk e-mail client yang berbasis web, e-mail palsu juga sulit dibedakan dengan e-mail yang asli. Gambar 4 dan gambar 5 merupakan contoh e-mail asli dan e-mail palsu yang diterima oleh *mail client* berbasis web dari e-mail yahoo :

Gambar 4 : E-mail asli dari yahoo



Gambar 5 : E-mail palsu dari yahoo

Kedua e-mail diatas terlihat sama. Untuk mendeteksi apakah e-mail tersebut asli atau tidak ada beberapa cara, yaitu :

- Melihat *header* email. Dengan melihat *header* e-mail, maka bisa diketahui rute e-mail tersebut dikirimkan. Namun cara ini sangat jarang dilakukan,

karena memang tidak tahu apa itu header dan bagaimana cara menganalisanya, sehingga penerima e-mail mudah tertipu dengan email palsu.

- Dengan menggunakan *digital signature*. Mekanisme ini juga jarang dilakukan karena tidak banyak orang yang menggunakan digital signature.
- Administrator harus rajin membaca log untuk melihat keanehan atau anomali dengan penggunaan email. Misalnya dilakukan pengamatan apakah ada orang mengirimkan email dengan identitas (From:) yang tidak sama dengan domain organisasi. Selain itu server e-mail juga harus dibatasi agar tidak ditumpangi oleh orang yang tidak berhak.

Penyaringan e-mail palsu

Keaslian e-mail yang kita terima dapat dideteksi dengan mempergunakan salah satu dari 3 (tiga) cara diatas. Pada penelitian kali ini, akan mempergunakan cara pertama, yaitu mendeteksi e-mail palsu berdasarkan proses validasi yang dilakukan pada header *field*. Perbedaan e-mail palsu dan asli, biasanya terletak pada domain asal mail server. E-mail yang asli akan memiliki domain mail server atau MTA yang sama dengan domain asal/identitas e-mail. Sedangkan e-mail yang palsu akan memiliki perbedaan antara domain mail server atau MTA dengan domain asal/identitas e-mail. Sehingga untuk mendeteksi e-mail asli atau palsu, dilakukan dengan cara ekstraksi header email yang selanjutnya header tersebut diekstraksi lagi untuk memperoleh domain server mail/MTA dengan domain asal/identitas e-mail. Gambar 6 dan gambar 7 contoh header e-mail asli dan palsu :

```
From y3dips Wed Nov 19 18:13:01 2003
X-Apparently-To: talent_spidey@yahoo.com via 66.218.78.66; Wed, 19 Nov 2003
18:11:12 -0800
Return-Path: <y3dips@plasa.com>
Received: from 202.134.0.35 (EHLO out-mta1.plasa.com) (202.134.0.35) by
mta101.mail.scd.yahoo.com with SMTP; Wed, 19 Nov 2003 18:11:10 -0800
From: <y3dips@plasa.com>  Add to Address Book
Subject: ini email aslinya
To: talent_spidey@yahoo.com
X-Mailer: CommuniGate Pro WebUser Interface v.4.1.6
Date: Thu, 20 Nov 2003 09:13:01 +0700
Message-ID: <web-8228099@b2.c.plasa.com>
MIME-Version: 1.0
Content-Type: text/plain; charset="ISO-8859-1"; format="flowed"
Content-Transfer-Encoding: 8bit
Received: from HELO b2.c.plasa.com by out-mta1.plasa.com with esmtp id
1AMeHj-001a3m-DM
Content-Length: 328


ini email asli nih


SALAM
====
y3dips
```

Gambar 6 : Header e-mail asli

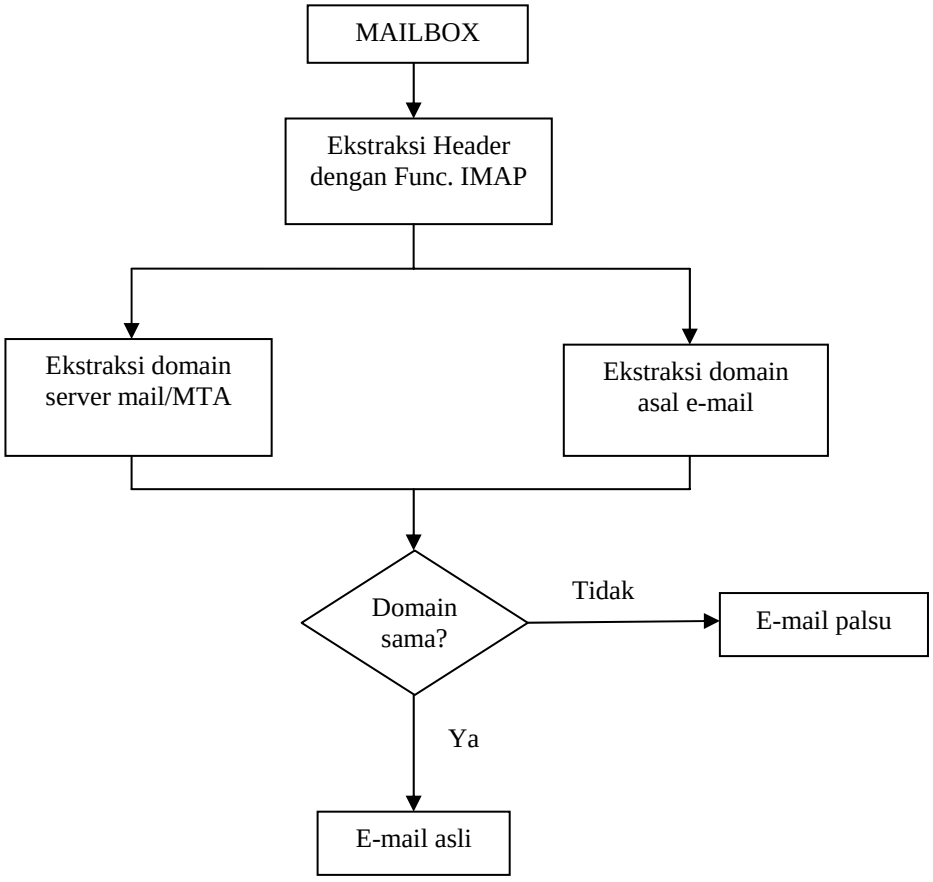
```
From y3dips@plasa.com Wed Nov 19 18:31:14 2003
X-Apparently-To: talent_spidey@yahoo.com via 66.218.78.65; Wed, 19 Nov 2003
18:33:04 -0800
Return-Path: <anonymous@hostcorporate.com>
Received: from 69.41.231.186 (HELO 10.hostcorporate.com) (69.41.231.186) by
mta113.mail.sc5.yahoo.com with SMTP; Wed, 19 Nov 2003 18:33:03 -0800
Received: (qmail 30147 invoked by uid 33142); 20 Nov 2003 02:31:14 -0000
Date: 20 Nov 2003 02:31:14 -0000
Message-ID: <20031120023114.30146.qmail@10.hostcorporate.com>
To: talent_spidey@yahoo.com
Subject: ini email palsunya
From: y3dips@plasa.com  Add to Address Book
Reply-to: y3dips@plasa.com
Content-Length: 42


ini palsu !! :P


SALAM
```

Gambar 7 : Header e-mail palsu

Dari gambar diatas dapat dilihat bahwa pada e-mail asli, tidak ada perbedaan domain asal e-mail dan domain mail server/MTA. Sedangkan pada e-mail palsu, terdapat perbedaan domain asal e-mail dan mail server/MTA. Secara garis besar, proses ekstraksi header dan identifikasi e-mail palsu dan asli dipresentasikan dalam diagram alir berikut ini :



Gambar 8 : Diagram alir Ekstraksi dan validasi header e-mail

Proses ekstraksi header e-mail, dilakukan dengan mempergunakan fungsi IMAP yang ada pada *script* PHP. E-mail yang sudah masuk dalam MAILBOX, diseleksi satu persatu dan headernya diekstrak. Berikut ini fungsi IMAP pada script PHP yang dipergunakan untuk mengekstrak header e-mail :

```
$header = imap_header($mailbox,
[nomor email]);
$vfrom = $header->fromaddress;
$idnya = $header->message_id;
```

Variabel $header berisi tentang header e-mail yang akan diekstrak. Variabel $vfrom untuk mengambil domain asal e-mail, sedangkan variabel $idnya untuk mengambil domain server mail/MTA.

Setelah header e-mail diekstrak, langkah berikutnya adalah proses ekstraksi domain asal e-mail dan proses ekstraksi domain server mail/MTA. Berikut ini fungsi IMAP pada script PHP yang dipergunakan untuk mengekstrak domain asal e-mail dan domain mail server/MTA :

```
//mencari domain asal e-mail
dari header from
$frmarray = explode("@",$vfrom);
$jfrar=count(explode(".",$frmarr
ay[1]));

//mencari domain server mail/MTA
$idarray = explode("@", $idnya);
$eidex=explode(".",$idarray[1]);
$jidar=count($eidex);
```

Pada script bagian yang atas, berfungsi untuk mencari domain asal e-mail yang diambil dari *field* **header from**. Fungsi **string explode** berfungsi untuk memecah field from e-mail, sehingga akan diperoleh domain asal e-mail.

Script yang dibawahnya berfungsi untuk mencari domain mail server/MTA. Dengan cara yang sama dengan yang dipergunakan untuk mecari asal e-mail (mempergunakan fungsi string explode), maka domain mail server/MTA dapat diperoleh dengan mengekstrak field header Message-ID.

Setelah diperoleh kedua domain asal e-mail dan server mail/MTA, maka kedua domain ini dibandingkan, sehingga akan diperoleh informasi apakan e-mail tersebut asli atau palsu. Berikut ini script PHP yang dipergunakan untuk seleksi domain tersebut :

```
if ($jfrar==$jidar){
   if ($frmarray[1] ==
$idarray[1])
      print("<p>E-mail ini tidak
palsu");
   else
      print("<p>E-mail ini
palsu");}
else{
   $idarray2 = "";
   for($i=($jidar-
$jfrar);$i<$jidar;$i++)
      $idarray2 =
"$idarray2"."$eidex[$i].";
   if ($frmarray[1] ==
rtrim($idarray2,"."))
      print("<p>E-mail ini tidak
palsu");
   else
      print("<p>E-mail ini
palsu");}
```

KESIMPULAN

Dari penelitian diatas, dapat disimpulkan beberapa aspek tentang keamanan e-mail khususnya yang berhubungan dengan e-mail palsu :

1. Administrator harus mengamati *file log* secara berkala untuk mengetahui adanya user dari luar yang mempergunakan mail server untuk membuat dan mengirim e-mail palsu
2. Konfigurasi mail server yang benar, akan mmencegah dan mengurangi kesempatan user untuk membuat dan menyebarkan e-mail palsu.
3. Penerima e-mail harus selalu waspada jika menerima e-mail yang isinya sangat penting dengan melihat header e-mail
4. Teknik yang dipergunakan untuk mendeteksi e-mail palsu masih perlu dikembangkan, sehingga hasil deteksi dan filtering lebih akurat.
5. Perlu dikembangkan teknik mendeteksi e-mail palsu dengan mempergunakan bahasa penrograman lain, selain dengan *script* PH**DAFTAR PUSTAKA**

[1] Atkinson, L., 1999, " Core PHP Programming :using PHP to build dynamic Web sites", Prentice Hall.

[2] Bakken, S., S., Aulbach, A., etc., 1997-2001, "PHP Manual", PHP Documentation Group.

[3] Chalanset, N., Cahagne, O., 2000, "NOCC Web Mail", http://sourceforge.net/projects/nocc

[4] Crocker, David H., August 1982, "Standard For The Format Of Arpa Internet Text Messages", RFC 822.

[5] Eaves,S., and Martin,G., October 14, 1999, "Electronic Mail", http://www.whatis.com

[6] Freed, N., Borenstein, N., November 1996, "Multipurpose Internet Mail Extensions (MIME) Part One : Format of Internet Message Bodies", RFC 2045.

[7] Palme, J., July 2002, "Common Internet Message Header Fields", Stockholm University.

[8] Raharjo, B. 2005, "Keamanan Sistem Informasi Berbasis Internet Versi. 5.3.", PT Insan Indonesia - Bandung & PT INDOCISC – Jakarta

[9] Resnick, P., April 2001, "Internet Messages Format", RFC 2822, QUALCOMM Incorporated.

[10] Vreuls, J., February 2, 2002, "eCorrei A webbased E-mail solution", http://ecorrei.sourceforge.net/