Mechanisms of processing e-mail on Linux mail servers: reducing security risks of using html code in e-mail

T. Vrbanec, B.S.
Teachers College in Čakovec
Ante Starčevića 55, Čakovec, Croatia
Phone: +385 (0)40 370 030
Fax: +385 (0)40 370 025
E-mail: tedo.vrbanec@vus-ck.hr

D. Radošević, Ph.D.
Faculty of Organization and Informatics
University of Zagreb
Pavlinska 2, Varaždin, Croatia
Phone: +385 (0)42 213 777
E-mail: danijel.radosevic@foi.hr

Professor Ž. Hutinski, Ph.D.
Faculty of Organization and Informatics
University of Zagreb
Pavlinska 2, Varaždin, Croatia
Phone: +385 (0)42 213 777
E-mail: zeljko.hutinski@foi.hr

Abstract - The widespread and rapidly growing use of html format of e-mail is accompanied with the presence of the malicious code (viruses, worms, active scripts, trojans, redirection to malicious web contents, and other similar things), especially in the light of the widespread illegal advertising. Apart from the antivirus tools and the tools used to remove spam, on Linux mail servers there are other tools like *Procmail, Maildrop, CSMail* and *MIMEdefang*, which can also be used to filter e-mail. In this paper the authors have shown a method for the development of the programme solution which can be added to the above mentioned tools. By using the suggested method the html contents is sorted out from the messages by putting it into the compressed attachment and thus leaves the pure text in the message. The user has thus been secured from the above mentioned threats and can decide for himself whether he wants to open the attachment or not.

I. INTRODUCTION

Transference of electronic messages requires (mail) servers which complete their tasks through SMTP, POP3 and IMAP protocols. The majority of mail servers in use today can be placed under one of two categories: Linux or Windows. It is hard to asses their percentage of the market. According to the *Yeenky Group* web surwey [1], *Microsoft* assesses it has an 85% market share of servers of the so-called small and medium business subjects, with projected 65% increase of sale of Windows servers in 2005. On the other hand, according to the same source, rate of growth of new Linux servers on the market is constantly increasing, from 15% in 2001 to 40% of total number of new servers in the world in 2004. Today, the number of new Linux and Windows servers is almost equal. Nevertheless, [2] Linux is gaining advantage in multiprocessor servers and in organizations with the need for cluster operations of the servers.

Based on the research made between December 2002 and April 2003 [3], we can make synthesis of tendencies of usage of individual operational systems for mail servers (Figure 1). We should emphasize the fact that the sample of mail servers included in the research in December 2002 is not representative, although in the case of mail servers it is hard to say how extensive the sample would have to be to be called representative, i.e. what is the total number of mail servers in the world.

According to the study of *Radicati Grupe* [4], 651 millions of people in the world use e-mail regularly, and this number will, according to the same source, increase to 850 millions by 2008.

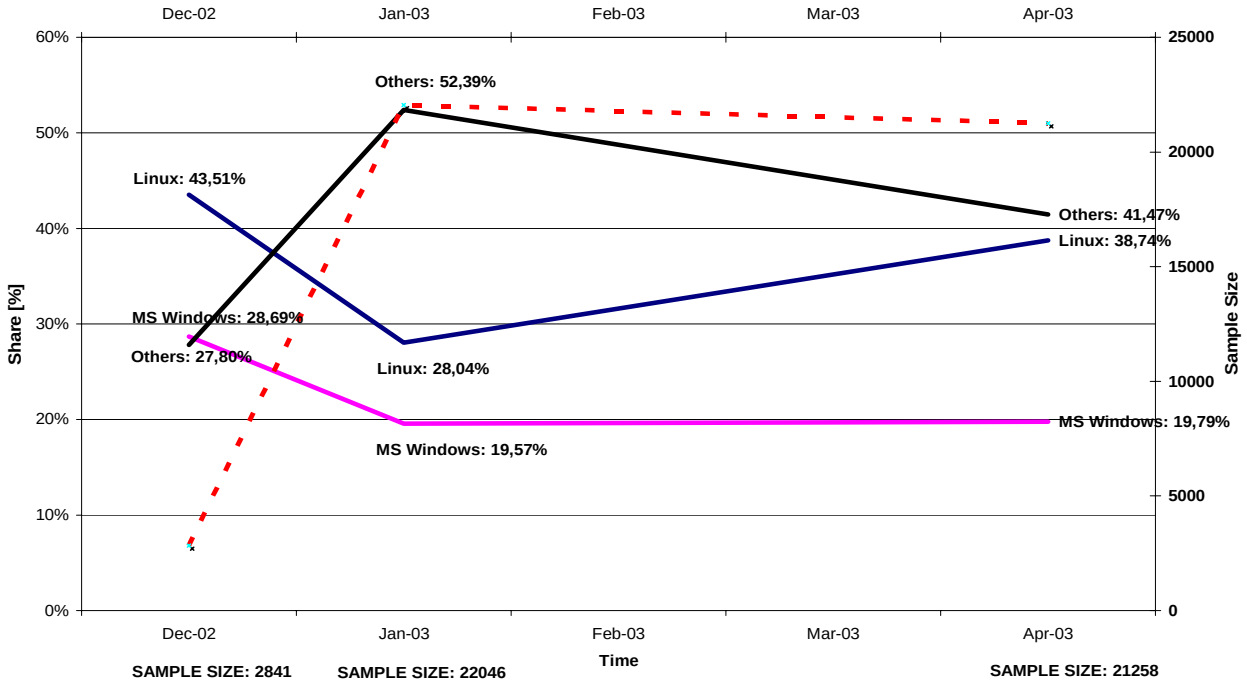

Figure 1 *MAIL Servers Operating Systems*

According to [5], the traffic of digital data doubles every year, while according to Moore's law, the power of processors doubles "only" every 18 months. This lag is being compensated mostly by increase of the number of servers. Unfortunately, the researches like [1] show that the percentage of unwanted e-mail in total number of e-mails in 2004 was, according to different sources, between 38% and 88% of all e-mails, while 7% of e-mails are infected.

According to [6, 7], the so-called "phishing" of user's accounts and financial data through e-mail during first half of 2004 increased by monthly rate of 50%. Some 70% of users received phishing e-mails, and some 5-15% was successfully deceived, i.e. they disclosed their confidential user and/or financial data. According to the same source, company *MessageLabs* argues that the number of variants of such e-mails increased in just six months from 279 to 215643. According to [8], the defense consists of powerful web and mail authentication and of digital signature of e-mails. We should also mention that FBI [8] considers phishing to be the most important new threat on the Internet (July 2003.).

Mail servers can successfully function with installed OS and Mail Transport Agent like *Sendmail*. Many of them have other useful functions, like filtering of unwanted emails or checking emails for some kind of infection. It depends upon the number of users of that server, and upon limitations of time, equipment, finances and educated system administrators.

Although the two main contestants in "market race" have different characteristics of safety of operative systems and applications, as well as different time of response to omissions in their codes, in this paper we will not address this issue. Nevertheless, we have noticed the three main reasons the users increasingly rely on Linux mail servers. In this, we consider the objectivity or subjectivity of these reasons perceived in the moment of decision about platform for mail server to be irrelevant:

- Perception of lower total costs
- Confidence that the program support of open code is manufactured with fewer security omissions
- Shorter time needed for solving possible security omissions.

II. E-MAIL THREATS

In the beginning of the development of the Internet, while the interchange of e-mail was in its early stages, the one and only form it used was pure text. Attachment and html forms appeared later. Many users from this age, as well as their students, shrink from the use of html form of e-mail. Some are so extreme in this that they do not use e-mail clients that support html form of e-mail. On the other hand, users who write e-mails exclusively in html form are mostly unaware of the fact that they received message in the form of pure text. To be compatible with the whole Internet community in relation to e-mail means to write it in form of pure text. Unfortunately, most of today's clients for sending and receiving e-mail have related settings on html, and it is possible to put malicious code within any html code, to make redirection to malicious code on the Internet, or to the fake web pages where deceit is used to extract confidential data from credulous or incautious users. It is possible [9, 10] that the future will bring some change, since *Microsoft* itself, as the world's largest manufacturer of operational systems and office packages, seems to be aware of the dangers related to it, and proposes to its users the change of settings which transforms html parts of e-mails into pure text.

Many users do not have rights for change of the settings of applications they start. Some use clients for e-mail reading and sending which cannot turn off sending of html forms of e-mails [11].

There are many dangers for e-mail recipient:

- Viruses
- Worms
- Trojans
- Malicious code (active scripts, program accessories – parasite code)
- Redirection to malicious Internet contents
- "Phishing" of confidential user data, mostly financial
- Web bugs
- Programs which change user's dial up settings.

All threats stemming from html form of e-mail can be divided into two groups of reasons:

1. html code can be executed, i.e. html code is, in broader sense, executive code (it can contain code in java script, VBscript, etc.)
2. html code can endanger the user's privacy.

There are no definite methods of user's self-defense, except that he/she defines sending and reading of messages in form of pure text in e-mail settings [12].

On various web sites with very different informative functions, it could happen that the data we received from them are not the data which that web site should have sent. Namely, if we follow the specially shaped link which could also be present in html part of e-mail, this web site, i.e. server can send wrong data, unwanted images and programs, as well as the malicious scripts whose effect is compromise of users' data [13].

Some of the ways in which the user can be exposed to the influence of malicious scripts are:

- Following of the links to the web sites to which we do not have full confidence, e-mails or posts on news groups.
- Usage of interactive forms on web sites which we do not trust completely.
- Review of dynamically generated sites whose content or author is unknown.

In case of malicious script usage, the attacker may get the users password or some other relevant information. Besides, malicious script can be used for endangering the local network, i.e. shared resources, or attacks on other computers or servers.

Some web browsers have vulnerable security system, which means that they decide which rights a script should have for it to be executed on the client computer. In such way, the scripts can be used to load and install various program packages, even to read or modify data on other web sites/servers. Malicious scripts can also be used for changing the review of web browser, to make phishing attacks more efficient through use of social engineering. For example, malicious script can open a window of the

browser outside of the visible field of screen, or lay a false address over field of address band. The raider can use malicious script of infection of web cookies by his/her copy. If the infected web cookie is sent back to the vulnerable web site and forwarded to the user's web browser, malicious script can be executed again. We should emphasize here that it is not the vulnerability of web cookies, but the malicious script implements the functionality of web browser for usage of web cookie. Therefore the whole concept of usage of web cookies has become very dubious.

Here is very educative and recent example of existing threats from the Internet. It happened on November 20th 2004 [14].On that day, a very popular server was compromised, and a malicious script was placed on its web sites: *Exploit.HTML.Iframebof*. When a user of Internet Explorer (except those from Windows XP with SP2) visited the sites of that server, the above mentioned *exploit* installed *Trojan-Downloader.Win32.Small.aaq*, which in turn tried to download and start *Backdoor.Win32.Agent.ec*, after which the user's computer was placed under control of the raider. During the same month, many other servers were successfully attacked. They directed the users to the web sites from which *adware*, *spyware*, and/or *porno dialers* were installed onto computer-victims. The computers infected in such manner were used for spreading unwanted e-mails or phishing attacks.

Although it is hard to understand the motives and criminal impulses of virus writers (except if some percentage of the viruses is not manufactured by the very same companies which make anti-virus programs), new viruses or their variants appear almost daily, causing great damages to the organizations. Even with the best anti-virus protection, it is always possible that some virus infects the computers of a local network in early stage, when the anti-virus protection of that network, i.e. organization, is not yet updated with a new component which would protect it from new virus. Here it is especially important to use multiple anti-virus protection of mail servers, which is made according to the financial resources of organization and burdening of the hardware, especially servers. Although there are many anti-virus tools which function excellently and timely, they have specific features related to the speed of response to the new viruses, worms or other malicious e-mail, and related to the possibility of discovering non-viral forms of threats.

In most cases, worms do not change the data on the infected computer and do not delete them (we cannot rely on this), but they often compromise them by sending them randomly to the addresses from user's address books, or by generating them through various algorithms. Sometimes they open the way to the more dangerous viruses or trojans, and some of them create enormous traffic through Internet, choking it in the process.

Trojans do not cause immediate damage. Sometimes the users are not aware of the existence of a trojan for a long time. Still, they "open the gates" of the computer, enabling the unknown raider to take control over the computer system or some part of it, to change or delete data, to add new data, to read the existing information without authorization, or, if the position on Internet enables it, to turn it into a center for illegal activities (*warez* servers or *zombi* computers needed for DDoS raids). DoS (*Denial of Service*) [15] raid is an attack whose result is denial of service by intentional generation of large quantities of net traffic which chokes network equipment and the servers, which become so burdened that they are not able to process legitimate traffic. The consequence of such action is that the legitimate users are not able to use network services DDoS (*Distributed Denial of Service*) is a form of raid by denial of services in which the sources of choking net traffic are distributed on Internet. "Most often, these are the computers which were previously broken into, so they can be used for the attacks on other networks or computers on Internet" [15].

New ways of misuse of (other people's) computers are constantly being discovered. During the installation of seemingly free of charge program support, a parasite program in the form of accessory or separate program product component is often installed. It taps into computer resources, and it can be used for advertising products or services, or as a way for illegitimate obtaining data about user's habits or data about his/her user names and passwords, about his/her credit cards; it may use its processing power for synchronized cluster mass calculation of myriad of computers on Internet, or for attacks on other computers. Most of those things happen without knowledge of computer user.

The content of some web sites on Internet cannot be approached without activating the possibility of starting active scripts, which sometimes install some type of malicious program code.

Many people do not want to receive e-mails in html form because they take more space in email inbox, and because of the possibility that the computer system is adjusted in such manner that html part of e-mail contains Internet images or web bugs [16, 17] for following the user visits or for evidencing that the recipient has read the e-mail. Web bug is a small, often invisible image added to web site, e-mail or some other document which can contain html code. Its purpose is to provide the companies which use them with insight into statistics of people reading the e-mail or web site. Web bugs are invisible because their size is mostly 1x1 pixel. Because of that, they are downloaded extremely fast and cannot be detected, except if they are different in color from the page they are placed upon.

III. MEASURES OF PROGRAM PROTECTION OF MAIL SERVERS

Since the new ways of endangering the safety of computer systems through the e-mail are constantly being developed, new security measures for protection of distribution system of e-mail are following them. Some of these measures should be implemented on every computer in the network, or any computer that enters in any way in data exchange with other computers. A part of these measures is specific for servers of all types, while part is specific for mail servers.

Measures of program protection for all computers:

- Selection of a stabile version of operational system
- Program updating of OS core
- Program updating of other, primarily basic application support
- Quality, updated and residual anti-virus protection in real time, with regular checking of all data bases of the computer

- Production of periodic safety copies of user data
- Use, protection and adequate choice of user passwords.

Measures of program protection of server:

- Installation of minimal number of program packages
- Installation and adjustment of program packages for automatic following of the changes in system
- Installation and adjustment of program systems for automatic detection of system intrusion
- Installation and calibration of operative system for closing the entrance ports of the server, (except those needed for services) together with installed and configured program systems
- Installation of program systems of daily detection of inadequate passwords of system users and their automatic informing.

Measures of program protection of mail server:

- Multiple anti-virus protection of all e-mail traffic, which also includes above mentioned non-viral forms of threats
- Protection from unwanted mail
- Installation of program systems for following statistics about number and types of infected e-mails
- Specialized program support.

One of possible examples of implementation of specialized program support is development of a program system that will prevent the dangers stemming from html code which is increasingly present in e-mail. Namely, e-mails written in such form are more easily infected than the e-mails written in pure text, and potential victims are also more easily deceived by e-mails written in such form.

IV. EXISTING SOLUTIONS FOR E-MAIL FILTERING

For mail servers implemented on *UNIX* and similar platforms, there are various program solutions for e-mail filtering. Among the most popular are:

a.) ***Procmail***. It is based upon the program language Perl. It enables making rules for filtering according to various criteria, as well as certain usual procedures of e-mail processing, like distribution according to various criteria, automatic informing of the sender or recipient of e-mail about detected unwanted contents and their size, format etc. Here is an example of such rule:

```
:0
* ? egrep -is -f $PMDIR/black.list
/dev/null
```

:0 is the beginning of the rule; ? is testing of the conditions; egrep writes lines according to the pattern; -i ignores the difference between capital and non-capital letters; -s prevents messages about mistakes; -f signifies *file*; $PMDIR signifies course to the folder with the data base black.list in which there is a list of patterns which should not be in the title of e-mail. If this condition is fulfilled, e-mail is irreversibly discarded by the precise definition of virtual folder /dev/null (which is not obligatory!).

b.) ***CSMail***. *Shareware* for Windows operating systems, especially for *.Net* technology, which removes viruses and stops unwanted mail. It is not an object of our interest because of incompatible OS for which it is intended, but it should be mentioned as a widespread program solution on Windows-based platforms.

c.) ***Maildrop***. It can change the existing MTA or complement it. It includes many advanced possibilities of filtering. If used independently, it is necessary to forward the e-mail according to the program. There are no significant differences in relation to *Procmail,* other than being much less widespread.

d.) ***MIMEdefang***. Program which questions, changes and filters e-mails, designed especially for *Sendmail*. It represents the interface between *Sendmail* and program for e-mail scanning, for example anti-virus programs. It presents complex, adjustable and robust solution for e-mail filtering. Its stability and reliability needs to be proven, but it promises the most of all the existing UNIX-oid solutions.

For *Microsoft Exchange* servers, such program systems, whose most visible parts are *Visual Basic* scripts, have existed for a long time. They are rare on *UNIX*-oid mail servers. There are some solutions, but their functionality is mostly only partial. For example, there are solutions that simply, but non-selectively remove html code from e-mail. Other transform some common *Microsoft* data base formats into form of pure text which is readable in any mail client or in any text editor. Some solutions simply deflect e-mails in html form, informing the sender automatically about it. This solution means that the automatic information tries to educate user, i.e. to explain the reasons for such action, but it can also cause dissatisfaction of the user who, rightly, deems that the non-infected e-mail should be delivered.

V. FEATURES OF THE SUGGESTED SOLUTION

While designing their own program solution for *UNIX*-id server, the authors took into account the following guidelines:

- All parts of e-mail should be preserved
- The access to the original data should not be complex (this is primarily related to the negative rating of a solution in which the data – potentially dangerous parts of e-mails – would be approached via http link placed in the body of the message)
- Additional increase of the message should be minimal
- The solution should not be demanding in terms of usage of processing time of the server or recipient of the e-mail
- Turning on the newly created security option should be optional, i.e. the user is provided with additional level of protection, but he/she has a right not to accept it
- Turning on and off of the additional security option should be as simple as possible
- Program solution should not endanger the existing system of work of mail server, i.e. it should not disable or reduce the efficiency of other levels and systems of e-mail protection

- Program solution does not process e-mail written in pure text form or e-mail with digital signature.

The system was implemented on Linux *Debian* server with *Sendmail* like MTA, for the server which is being actively used. All the above mentioned guidelines were followed. In fact, e-mails have become somewhat smaller. It was done in the following manner: html part of e-mail is compressed and put in e-mail like attachment, with addition of short informing note into the body of e-mail. The compressed attachment is smaller than the original html code, except with very small html parts. Compression deactivates malicious code, and in most cases decreases e-mail. If the user thinks there is no danger of malicious codes, he/she can reach the original message.

VI. DESIGN AND IMPLEMENTATION OF PROGRAM SOLUTION

Design of the program solution

The solution which needs to satisfy larger number of requirements needs work and cooperation of several modules. At the level of system, all starts with *Procmail*, whose original task is e-mail filtering. Two of its features are emphasized;

- Ability to call executive files of any kind, which provides the system with satisfying adaptability.
- The e-mail processing stops in the moment of fulfillment of any of the defined rules.

The other feature deserves more attention and explanation. The rule generally consists of the condition or sequence of conditions for one action. One such action is, naturally, moving e-mail into certain folder, but any other action is also possible. The cessation of e-mail processing as soon as one rule is realized is useful for keeping the resources of the server. Still, in case of need, cessation of the processing can be avoided through providing the rule with the help of logic operator for negation and the ability of *Procmail* to copy e-mail before its processing.

We will start with the external features of processing. The system named ASS (Advanced Secutity System), adds the following part into the message header:

"*X-Notice: Checked by Advanced Security System at vusck.hr*"

only if it establishes that the user wants to activate the system, i.e. if the user has modified the file .html in his/her $HOME folder to contain the key word "ON". Namely, the default value which the user acquires while opening user account is set to "OFF". Under the same condition, the system adds the following signature at the end of body message visible to the user:

"--

Checked by Advanced Security System at VUS

[Teachers Colledge in Cakovec, Croatia]"

The change of content of the mentioned .html file which is inherited as the default content of the $HOME user folder can be realized in four ways:

a) Through application of the user to the server and the change through some text editor [with the instruction which it acquires during the opening of the user account or the instructions from the web sites].
b) Through oral or written report to the system administrator which realizes the request instead of the user.
c) Through directing the e-mail to the system administration with certain key words in the e-mail subject.
d) Through the application of the user to the server and through initiation of some of the *htmlon* or *htmloff*, commands of the simple scripts which create or modify *.html* file.

In such way, the user himself/herself turns on the suggested mode of protection. In the future, the same *php* script can be made which would be included in the application for *webmail* as a module; through this application, the user would be able to change his properties using the web interface. It would be possible to introduce other parameters in practical manner.

If the user does not turn on the system, common e-mail text message comes through to the user's account without the processing. For every e-mail, the following flow diagram can be applied:

```
If the user wants the processing
  then
  {
  If not (e-mail has the digital signature or is encrypted)
        then
        {
        add the header
        add the signature
        make the compressed copy of the e-mail
        deconstruct e-mail to the integral parts
        If not (plain text part exists)
                then {transform html segment into text}
        }
        or else {let the message through into the user's
                account}
  }
  or else {let the message through into the user's
          account}.
```

The realization of the described procedure is implemented through:

- Above described *Procmail*,
- *Formail* (for manipulation with the segments of e-mails),
- *Mpack* (for deconstructing e-mail into the integral parts and for constructing e-mails from the integral parts),
- *Zip* (for the compression of the original messages),
- *Perl* (scripts for checking and processing, and numerous Perl moduls),
- *Linux shell scripts* (for creating the simple commands),

- *Lynx* (for transformation of html into text),
- System software as the base of e-mail server.

The model of e-mail processing is visible in the following figure.

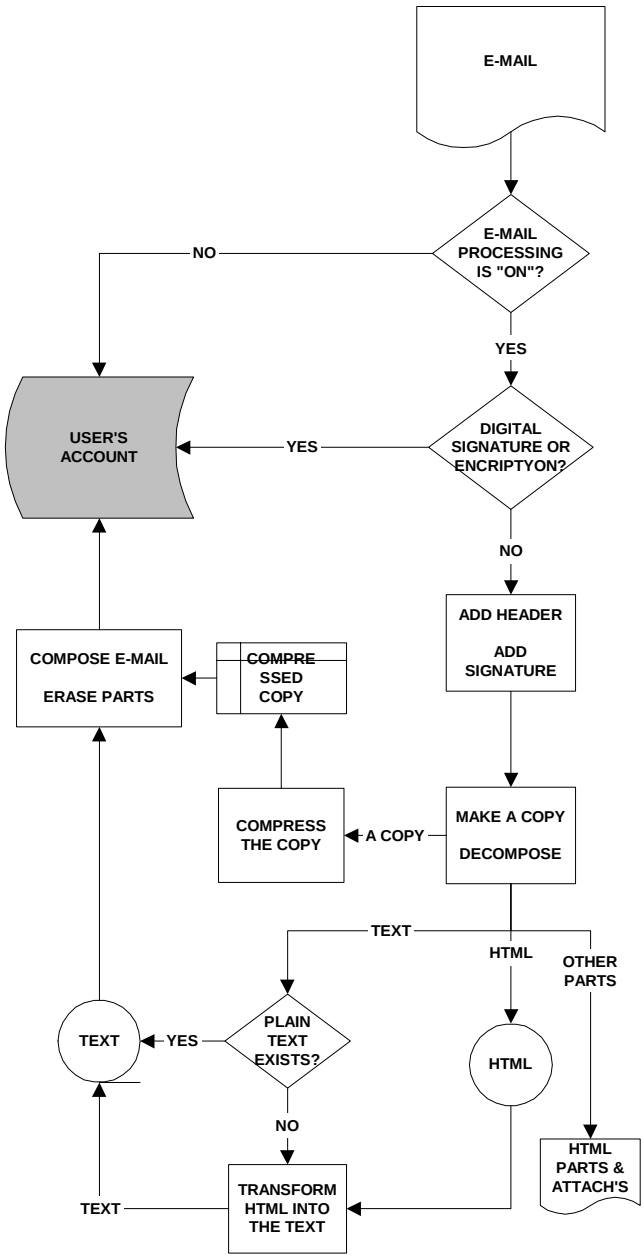

Fig. 2. Model of software solution

At the end of this presentation, it has to be mentioned that the actual implementation of the system is made in such way that every e-mail is subjected to multiple analysis through the anti-virus tools and through the self-learning tools for detecting and eliminating the unwanted messages. Finally, before it goes to the descried type of the process, it is subjected to additional filtering of the manually defined filters for unwanted mail introduced by the server administrator. However, every user can define the rules for himself/herself using *Procmail*.

The resulting e-mail has the following properties:

- Every part of the message is preserved in the compressed attachment, while the textual part of the messages visible without opening of the attachment,
- The access to the original data consists of the opening of the compressed attachment, which is simple action for the recipient. At the same time, the activation of the malignant code is disabled if the recipient decides not to open the attachment,
- Additional burdening of the message size is minimal, because compressed form is used; moreover, such e-mail is often smaller than the original,
- The user may decide for himself/herself whether he/she wants to use such form of protection or not.

Implementation of the program solution

Such system requires educated users, and the education relates to two categories:

1. Education about the possible threats from the Internet, especially in the domain of e-mail.
2. Education and training of the users for more than the basic use of ICT.

Such system is not a conventional one and it requires certain motivation of the user. According to the authors' experience, an average user, unfortunately, does not have intrinsic desire for new knowledge in the fields of security. Because of small number of the users who accepted this safety option of the system, the authors are not yet able to make comparative analysis of the effects of implementation of such system. Nevertheless, if the problem of the safety of e-mail dramatically escalates in some moment, it is possible to turn on the system of html processing, temporarily or continuously, as the "secret weapon" in the struggle for safety of the endangered users and resources of the organizational computer systems.

VII. CONCLUSION

"*Feature-rich email is not only a powerful way of communication, but also a major security threat.*" [18].

Every morally and ethically correct way of removing threats to the computer, user and his data is very important for the development and future of the Internet. We are faced with ever more serious and frequent threats with the Internet community; therefore we should implement every measure we have to reduce the danger to the users, their computers and data, as well as the distributed systems they may belong to. In this, we should be careful not to reduce the rights of the users, i.e. not to implement measures that would endanger privacy and authenticity of the data being exchanged through Internet. The methods of protection can be applied most efficiently in the **server** field. Within the domain of e-mail and mail servers, by implementation of their program solution, the authors contributed to the tendencies for the Internet to become not only the fastest means of communication and information, but also a **safe** place for its visitors.

REFERENCES

[1] IT Facts, *Categories: Servers*, <http://www.itfacts.biz/index.php?id=C0_5_1>, (January, 2005.)

[2] IDC - Press Release, *Linux Server Adoption Broadens, Dual-Processor Systems Becoming Predominant Form-Factor, According to IDC*, <http://www.idc.com/getdoc.jsp?containerId=pr2004_11_02_093312>, (January, 2005.)

[3] Credentia - digital security, email, web, dns & amp; wireless, *E-Mail Server Survey Results*, <http://www.credentia.cc/research/surveys/smtp/>, (January, 2005.)

[4] The Radicati Group Inc., *Q4 Market Numbers Update*, <http://www.radicati.com/pubs/news/Q4-2004_PressRelease.pdf>, (January, 2005.)

[5] In-Stat, *Covering the Full Spectrum of Digital Communications Market Research*, <http://www.instat.com/>, (December, 2004.)

[6] CNET News.com, *'Phishing' scams luring more users*, <http://news.com.com/2100-7355_3-5194807.html?part=rss&tag=feed&subj=news>; <http://news.com.com/Caught+in+a+phishing+trap/2100-1029_3-5453203.html?tag=st.rc.targ_mb> , (January, 2005.)

[7] CNET News.com, *Phishers learn new tricks*, <http://news.com.com/Phishers+learn+new+tricks/2100-7349_3-5544193.html?tag=st_lh>, (January, 2005.)

[8] The Anti-Phishing Working Group, *Proposed Solutions to Address the Threat of Email Spoofing Scams*, <http://www.antiphishing.org/>, (January, 2005.)

[9] Increase Your Browsing and E-Mail Safety: (4 Steps to Help Ward Off Hackers and Attackers), *Step 3: Read E-Mail Messages in Plain Text*, <http://www.microsoft.com/security/incident/settings.mspx>, (January, 2005.)

[10] Microsoft Hrvatska, *Description of a new feature that users can use to read non-digitally-signed e-mail or nonencrypted e-mail as plain text in Office XP SP-1*, <http://support.microsoft.com/default.aspx?scid=kb;en-us;307594&Product=ol2002>, (January, 2005.)

[11] ZZEE 1st Email Anti-Virus. Email content security utility: protects from HTML and MIME based mail threats, <http://www.zzee.com/enh/>, (January, 2005.)

[12] US-CERT, *Vulnerability Note VU#842160*, <http://www.kb.cert.org/vuls/id/842160>, (January, 2005.)

[13] CERT Coordination Center, *Frequently Asked Questions About Malicious Web Scripts Redirected by Web Sites*, <http://www.cert.org/tech_tips/malicious_code_FAQ.html>, (January, 2005.)

[14] Schouwenberg, R., Viruslist.com - *Servers compromised all over the globe*, <http://www.viruslist.com/en/news?id=155636876>, (January, 2005.)

[15] Č.P.P. + CERT.hr nacionalno središte za računalnu sigurnost, *Što je DoS napad? DDoS?,* <http://www.cert.hr/faq.php?lang=hr&id=17>, (January, 2005.)

[16] Lowe, R., C. Arevalo-Lowe, *Web Bugs*, <http://www.leave-me-alone.com/webbugs.htm>, 1999.-2002. (January, 2005.)

[17] Smith, R., M, EFF: *The Web Bug FAQ*, <http://www.eff.org/Privacy/Marketing/web_bug.html>, December 11th 1999. (January, 2005.)

[18] Slavic, P., *A quick guide to email security and what's wrong with a generic antivirus program*, <http://www.zzee.com/email-security/>, (January, 2005.)

[19] Netcraft, *January 2005 Web Server Survey*, <http://news.netcraft.com/archives/2005/01/01/january_2005_web_server_survey.html>, (January, 2005.)

[20] Secunia – Advisories, *Internet Explorer HTML Elements Buffer Overflow Vulnerability*, <http://secunia.com/advisories/12959/>, (January, 2005.)

[21] CNET News.com, Linux closing in on Microsoft market share, <http://news.com.com/2100-1001-243527.html?legacy=cnet>, (January, 2005.)

[22] Nemeth, E., G. Snyder, T. R. Hein, *Linux Administration Handbook*, Prentice Hall PTR, Upper Saddle River, 2002.