Comunicado Técnico 65

Março, 2005
Campinas, SP

ISSN 1677-8464

Uma Solução para Melhoria de Desempenho para Servidores de Correio Eletrônico com Antispam

Marcelo Gonçalves Narciso[1]

Existem *softwares* antispam que avaliam o cabeçalho e o corpo da mensagem e o classificam como spam ou não. Eles têm uma margem muito pequena de erro, menos de 0,5%. Os *softwares* mais conhecidos desta linha são: spamassassin (Spamassassin, 2004) , dspam (DSPAM, 2004), bogofilter (Bogofilter, 2004).

Um problema relativo aos softwares antispam citados é o fato de que eles sobrecarregam o servidor de mensagens pois toda mensagem que chega deverá ser analisada pelo *software* antispam antes de ser entregue, e esta análise demora um certo tempo não desprezível.

Outros *softwares* procedem de maneira diferente quanto a reter spams. São aqueles que usam do artifício de desafios fáceis para o remetente. Uma pergunta simples, um reply, etc. são exemplos dos desafios que este tipo de sistema pode fazer. O mais conhecido é o antispam da UOL (UOL, 2004) no qual o remetente lê uma palavra em uma imagem e responde uma pergunta com aquela palavra. Após isto, a mensagem é enviada ao destinatário. Este tipo de sistema também sobrecarrega o servidor, pois para cada mensagem enviada é disparado um processo.

Na Embrapa Informática Agropecuária, os *softwares* utilizados são spamassassin (Spamassassin, 2004), que representaria o comportamento dos filtros classificatórios (Bayes, 2004) e o TMDA que representa um antispam do tipo desafio/resposta. Este último foi utilizado com pouco usuários, mas foi possível verificar que também pode sobrecarregar o servidor se muitos processos forem disparados. Este trabalho descreve a experiência obtida com estes filtros antispam relativos à influência destes sobre desempenho do servidor e como otimizá-lo ao máximo, usando o entregador de mensagens locais, *procmail*, e seu arquivo de configuração *procmailrc*.

Procmail

Quando o *software* relativo ao servidor de correio eletrônico (*sendmail*, *postfix*, exim, *qmail*, etc.) reconhece que a mensagem deve ser entregue a um usuário do domínio local, é chamado um programa conhecido como local *mailer* (entregador de mensagens locais). Um possível programa para este fim é o *procmail* (Procmail, 2004).

O modo de funcionamento do *procmail* é direcionada pelo arquivo de configuração *procmailrc*. Este arquivo é usado para configurar a forma de entrega da mensagem para um usuário local e pode haver diretivas para que as mensagens passem por filtros para então serem entregues. Estes filtros podem ser usados para verificar extensões de arquivos anexos, conteúdo do arquivo ou campo do título da mensagem, etc. Além disso, filtros antispams podem ser configurados para avaliar a mensagem antes da mesma ser entregue.

Além de ser usado para reter spams, o *procmail* tem outras aplicações interessantes, tais como barrar arquivos anexos a mensagens que podem ter vírus (Narciso, 2001).

[1] *Doutor em Computação Aplicada, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: narciso@cnptia.embrapa.br)*

Embrapa

Softwares Antispams Classificatórios

Estes tipos de *softwares* analisam o corpo e o cabeçalho de uma mensagem e usam a idéia do teoremas de Bayes para classificar uma mensagem. *Softwares* como Bogofilter (Bogofilter, 2004), Dspam (DSPAM, 2004) e recentemente o spamassassin são exemplos deste tipo de filtros. Mais detalhes podem ser vistos em Bayes (2004). A vantagem deste tipo de filtro é que ele é adaptativo, isto é, à medida em que as técnicas dos remetentes de spams vão mudando, este tipo de filtro aprende as mudanças e continua a combater os spams. Para isto, o *software* é treinado com uma amostra de spams (geralmente 1.000 ou mais) para poder então continuar atuando eficientemente.

Neste trabalho, foi usado o spamassassin, *software* mundialmente conhecido. Além de ser um filtro antispam bayesiano (versões mais recentes), o spamassassin também usa um conjunto de regras para verificar se a mensagem é um spam ou não. Este conjunto de regras sempre se atualiza e pode ser obtido a partir do site (Spamassassin, 2004).

Softwares Antispams do Tipo Desafio/Resposta

Existe uma outra categoria de *softwares* antispams que agem de forma a certificar que o remetente é válido e, desta forma, verificar se a mensagem é de origem confiável. Para isto, este sistema, ao receber uma mensagem, envia uma outra mensagem para o remetente com a finalidade de desafiá-lo a responder uma pergunta ou tomar uma outra ação que seja fácil. Caso o remetente responda corretamente, a mensagem original é entregue ao destinatário final. A seqüência seria:

1. o remetente envia mensagem para o destinatário;
2. o servidor de correio eletrônico retém a mensagem e envia uma outra ao remetente desafiando-o a responder alguma pergunta ou similar;
3. o remetente responde a mensagem e esta vai para o servidor de correio eletrônico;
4. se a resposta dada pelo remetente for correta, a mensagem original é entregue ao destinatário. Se a resposta estiver errada, a mensagem é descartada.

O TMDA (TMDA, 2004) é um software da categoria desafio/resposta. O desafio que o servidor envia ao remetente é que este responda à mensagem (reply to) simplesmente. Assim, o programa que envia spam provavelmente não irá responder à pergunta. Este software tem a vantagem de evitar perda de mensagens legítimas (false positive). Porém, a entrega de mensagens, quando o remetente não está na lista branca, demora alguns minutos, desde que o remetente responda o desafio.

O TMDA pode ser usado em conjunto com o spamassassin e demais filtros. A configuração é feita no arquivo *procmailrc*. Mais detalhes podem ser vistos em (TMDA, 2004)

Solução Proposta

Os softwares antispam devem ser os últimos a serem invocados pelo sistema de entrega de mensagens, após outros filtros mais simples que retenham spams sejam executados, devido à questão de desempenho do servidor. Assim, o menor conjunto possível de mensagens será analisado pelos *softwares* antispam, diminuindo a carga sobre o servidor, visto que os *softwares* antispam consomem CPU de forma considerável. Para isto, o sistema de lista negra do servidor deve estar sempre atualizado e deverá estar sempre bem configurado. A maior parte dos spams são rejeitados desta forma, conforme pode ser visto pela Tabela 1.

Tabela 1. Resultados de mecanismos de retenção de spams.

Dia/abril	*Lista negra interna*	*Endereços falsos*	*Spamassassin*	*Filtros feitos em procmailrc*	*Total*
20	2.003	6.813	955	49	9.820
21	2.522	4.950	692	55	8.219
22	2.797	5.590	869	30	9.286
23	2.838	5.712	1.004	43	9.597
24	2.800	5.303	644	15	8.762
25	2.442	4.773	627	35	7.877
26	2.365	5.304	831	80	8.580
27	2.414	5.323	878	61	8.676
28	2.838	5.712	1.004	43	9.597

Observe que a lista negra interna que o servidor contém e os endereços falsos são os responsáveis pela grande maioria das mensagens retidas. Os *softwares* usados como Mail Transfer Agents - MTA tais como *sendmail*, *postfix*, *qmail*, exim possuem mecanismos de lista branca ou negra, que deixam passar ou retém a mensagem. Quando uma mensagem chega, o servidor analisa o domínio de onde a mensagem vem. Se o domínio não existir, a mensagem é descartada. Na Tabela 1, tem-se uma amostra das mensagens que são retidas.

Apesar da grande quantidade de mensagens retidas pela lista negra e pelo fato do domínio do remetente ser inexistente, ainda existe uma grande quantidade de mensagens que passam e devem ser analisadas. Desta forma, outros mecanismos devem ser usados para filtrar as mensagens que restam. Os filtros podem ser colocados no arquivo *procmailrc*, arquivo de configuração do *procmail*. Neste trabalho a ênfase será sobre o *procmail* (e *procmailrc*) pois este mailer local pode ser usado por qualquer MTA, sendo assim mais abrangente.

Configuração de Filtros em Procmailrc para evitar Spams

O arquivo *procmailrc* pode conter regras para diversas finalidades. O executável *procmail*, quando da entrega da mensagem, lê o arquivo *procmailrc* em ordem seqüencial. É interessante que as regras estejam na seguinte ordem, conforme experiência obtida com este *software:*

1. regras para barrar anexos de mensagens com extensões do tipo .exe, .bat, .pif, etc. para evitar disseminação de vírus. Como os vírus são feitos antes das vacinas, é interessante ter este mecanismo durante o tempo em que o vírus é descoberto e a vacina é feita. Algumas horas de atraso para se fazer a vacina são suficientes para contaminar muitos *sites.* Este mecanismo é de suma importância, mais até do que filtros antispam;

2. verificar se a mensagem com o domínio do *site* vem mesmo de algum usuário do *site.* Se vier, entregar. Caso contrário, rejeitar;
3. inserir lista branca para *sites* conhecidos e lista branca específica sob demanda dos usuários;
4. inserir filtros sobre frases ou palavras no título da mensagem ou corpo, quando for o caso. É válido quando os *softwares* antispam não conseguem reter mensagens. Assim, palavras-chaves podem ser usadas para reter spams;
5. inserir os filtros antispam (dspam, spamassassin, tmda, etc.).

A respeito da regra 1, tem-se um exemplo de regra para reter arquivos anexos a mensagens conforme a extensão, isto é:

```
:0 HB
.Filename*=?*\".*\.(exe|png|com|vbs|pps|bat|mov|
mpg|pif|js|shs|scr|chm|dll|hta|bas|lnk|isn|ade|
Adp|cmd|cpl|crt|hlp|inf|ins|isp|jse|mdb|mde|msc|
msi|msp|mst|pcd|reg|sct|url|vb|vbe|wsc|wsf|shb|
shl|htt|sc|zip|wsh)(=\?)?\"
| /etc/procmail.d/recusanexo.pl
```

O *script* recusanexo.pl envia uma mensagem de que o arquivo foi retido, caso o anexo tenha alguma das extensões citadas. Mais informações podem ser vistas em Narciso (2001) com detalhes.

Um dos meios que os remetentes de spam usam para conseguir seus objetivos é forjar uma mensagem com o remetente tendo o mesmo domínio que o destinatário. Assim, a mensagem provavelmente deverá passar pois não se encontra na lista negra do servidor e o domínio é válido. Para evitar este tipo de spam, é necessário construir uma regra, a qual verifica se o ip do domínio forjado corresponde realmente ao endereço que a mensagem sugere. Na maioria dos casos, o ip não tem como ser forjado pois o servidor que envia a mensagem escreve seu ip real. Uma possível solução para verificar se o domínio é correto ou não é usar a seguinte regra em *procmailrc.*

```
# Se a mensagem não vem de um ip interno (200.0.70.0. ou 192.207.194.0
# ou 127.0.0.1) e tem o domínio cnptia.embrapa.br então recusar

:0
* ^From:.*cnptia\.embrapa\.br
  {
     :0
     * !^Received: from[ ].*\[200\.0\.70\.
     * !^Received: from[ ].*\[192\.207\.194\.
     * !^Received: from[ ].*\[127\.0\.0\.1
     | /etc/procmail.d/falseUser.pl
  }
```

O *script* feito em perl, /etc/*procmail*.d/falseUser.pl, é executado se o usuário tem endereço do tipo xxx@yyy.cnptia.embrapa.br e o endereço ip do site de onde o remetente enviou a mensagem não corresponder aos endereços ip possíveis do domínio cnptia.embrapa.br. Este *script*, basicamente, envia uma mensagem ao remetente notificando o porquê da recusa da mensagem. Este *script* pode ser feito em qualquer linguagem que o usuário dominar.

Se a mensagem é realmente do domínio cnptia.embrapa.br, a mensagem não é retida e é entregue ao destinatário. Observe que assim se evita que a mensagem seja analisada pelos filtros antispam, evitando processamento desnecessário no servidor, visto que uma grande parcela de mensagens que passam pelo servidor é do próprio domínio, não necessitando assim ser analisado.

Na Embrapa Informática Agropecuária, esta regra barrou 100% das mensagens com este artifício. Só não consegue barrar se a mensagem for forjada de tal forma que o cabeçalho tenha o ip de origem da mensagem igual a um dos endereços ip relativos ao domínio cnptia.embrapa.br.

Seguindo a proposta de inserção de regras no arquivo *procmailrc*, tem-se a lista branca para evitar que mensagens de origem conhecida e confiável passe pelos filtros antispam e sejam retidos. Com isto, evita-se que grande quantidade de mensagens seja analisada por filtros antispam. Para dar acesso a domínios, pode ser feito como segue:

```
# Se a mensagem vem do cnptia, então deverá ser entregue
:0
* ^From:.*cnptia\.embrapa\.br
  {
     :0
      * ^Received: from[ ].*\[200\.0\.70\.
     $DEFAULT
     :0
     * ^Received: from[ ].*\[192\.207\.194\.
     $DEFAULT
     :0
     * ^Received: from[ ].*\[127\.0\.0\.1
     $DEFAULT
  }
```

Um outro exemplo, que seria uma lista branca, seriam mensagens que vêm da Fapesp.

```
:0
* ^From:.*fapesp\.br
  {
     :0
     * ^Received: from[ ].*\[143\.108\.10\.
     $DEFAULT
   }
```

Observe que um pedaço do endereço IP da Fapesp vai ser verificado, para garantir que a mensagem é mesmo do

domínio fapesp.br. O mesmo foi feito com o domínio cnptia.embrapa.br. Caso a mensagem seja proveniente do domínio fapesp.br, por exemplo, e o ip que estiver no cabeçalho da mensagem conferir com o do filtro, então a mensagem é entregue imediatamente ao destinatário (opção $DEFAULT).

Caso a mensagem seja forjada, ela não será entregue imediatamente e será avaliado pelo filtro antispam, ou ainda, pode ser feita outra regra para rejeitá-lo, tal como foi feito para o caso do domínio cnptia.embrapa.br, mostrado anteriormente.

Assim, grande parte das mensagens que passam pelo servidor, que são provenientes do próprio domínio ou de domínios conhecidos, são entregues ao destinatário sem precisar passar pelo filtro antispam. Este processo foi validado na Embrapa Informática Agropecuária e pode ser adaptado para qualquer site.

Como próxima proposta de inserção de regras no arquivo *procmailrc*, tem-se os filtros por palavras-chave ou frases que caracterizam a mensagem como spam. A vantagem destes filtros é reter mensagens que o *software* antispam não consegue, ou ainda, impedir que a mensagem seja avaliada pelo *software* antispam e consuma recursos de CPU do servidor. Outra vantagem destes filtros é que as mensagens retidas por estes podem ser usados para treinamento do *software* antispam (bogofilter, dspam, spamassassin, etc.).

Como exemplos de filtros por palavras ou frases, tem-se aqueles que retém a mensagem se a mesma contiver viagra, ou Rolex watch, etc. Um exemplo relativo à palavra viagra está a seguir:

```
#A regra abaixo rejeita todo arquivo que tiver a
palavra viagra

BODY = "viagra"

:0 c

* B ?? viagra

| /etc/procmail.d/palavraChave.pl $BODY

:0 HB

* B ?? viagra
/Export/home/perl/spamTrein
```

No exemplo citado, se a palavra viagra estiver no corpo da mensagem, o *script* palavraChave.pl vai enviar uma mensagem para o remetente dizendo que a mensagem foi retida por ter a palavra viagra em seu corpo (observe que o remetente é de fora do domínio e não está em lista branca alguma). A seguir, a mensagem é armazenada no diretório /export/home/perl/spamTrein, podendo ser usada para treinar o spamassassin ou outro filtro bayesiano qualquer. Após uma série de filtros para reter spams, tais como o do exemplo citado, insere-se a seguir a chamada do filtro antispam (regra 5). A forma de se chamar cada filtro, usando-se o *procmail*, pode ser vista em (TMDA, 2004) e (Spamassassin, 2004).

Após todas estas considerações, foi possível reter mais spams e melhorar o desempenho do servidor. A Tabela 2 mostra os spams retidos recentemente com os filtros configurados no arquivo *procmailrc*.

Tabela 2. Resultados de mecanismos de retenção de spams.

Dia/ outubro	*Usuário falso*	*Regras por palavras-chave*	*Spamassassin*	*Total*
09 - sábado	44	333	1.300	1.677
10 - domingo	25	256	964	1.245
11- segunda	49	160	1.155	1.364
12- terça	41	269	1.188	1.498
13 - quarta	41	316	1.378	1.735
14 - quinta	32	292	1.671	1.995
15 - sexta	31	213	1.582	1.826
16 - sábado	6	194	1.294	1.494

Observe que os quesitos da Tabela 2, se comparados aos da Tabela1, levando-se em conta que o quesito *procmail* da primeira tabela seria a soma de "usuário falso" + "regras diversas por palavra chave", têm um melhor valor, isto é, mais spams foram retidos. Em abril quando foi elaborada a Tabela 1, poucos filtros existiam e a maior parte do processamento era por conta do *software* antispam, o que degradava o desempenho do servidor. Agora, como o arquivo *procmailrc* têm mais regras e também entrega as mensagems provenientes do mesmo domínio sem passar pelo *software* antispam, o desempenho do servidor melhorou, além de um número maior de spams serem retidos.

Conclusões

O desempenho do servidor de correio eletrônico melhorou após esta ordem de organização das regras ter sido implantada pois o número de mensagens que o antispam verifica é muito menor

O software TDMA não tem problemas com "false positive", mas como ele envia pergunta, recebe resposta, e em seguida envia a mensagem, toma mais processamento do servidor do que o spamassassin. Assim, é de suma importância que o TMDA seja usado o mínimo possível, daí o fato das listas brancas definidas no arquivo *procmailrc* e demais filtros serem de suma importância para não sobrecarregar o servidor.

Os filtros relativos a palavras-chave ou frases contribuem não só para evitar spams, mas também para realimentar o spamassassin, no que se refere aos mais retidos por estes. Estas mensagens, em boa quantidade, acima de 1.000, servirão para treinar o spamassassin e, desta forma, melhorar seu rendimento. Este foi um dos fatores pelos quais o número de spams retidos pelo spamassassin aumentou, conforme constatado na Tabela 2, em relação à Tabela 1.

Seja qual for o MTA que estiver sendo usado no servidor, as regras relativas ao *procmail* são sempre úteis, visto que pode ser usado como entregador de mensagens locais de qualquer MTA. Assim, é genérico e, com pouco esforço de configuração de regras antispam no arquivo *procmailrc*, pode melhorar consideravelmente o desempenho do Servidor, evitando chamadas desnecessárias a *softwares* antispam. Outro detalhe é que, se o *software* MTA for

mudado, não é necessário alterar nada, evitando perda de tempo que se gasta em aprender a configurar o novo MTA contra spams, além das demais configurações para o *site* em questão.

Referências Bibliográficas

BAYES. Disponível em: <http://www.propus.com.br/articles/alt/1/1.html>. Acesso em: 10 nov. 2004.

BOGOFILTER. Disponível em: <http://www.bogofilter.org>. Acesso em: 10 nov. 2004.

DSPAM. **SourceForge.net: project info** - DSPAN AntiSpan Agent. Disponível em: <http://sourceforge.net/projects/dspam>. Acesso em 10 nov. 2004.

NARCISO, M. G . **Instalação de antivírus na servidora de mail**: Uma opção para impedir ataques de vírus anexados a e-mail. Campinas: Embrapa Informática Agropecuária, 2001. 6 p. (Embrapa Informática Agropecuária. Instruções Técnicas, 4). Disponível em: <http://www.cnptia.embrapa.br/publica/2001/INSTR%20TECNICAS%204%20int.pdf>. Acesso em 10 nov. 2004.

PROCMAIL [homepage]. Disponível em: <http://www.procmail/.org>. Acesso em: 10 nov. 2004.

SPAMASSASSIN. Disponível em: <http://www.spamassassin.org>. Acesso em: 10 nov. 2004.

TMDA. **Tagged Message Delivery Agent (TMDA)**. Disponível em: <http://tmda.net>. Acesso em 10 nov. 2004.

UOL - o melhor conteúdo [home page]. Disponível em: <http://www.uol.com.br>. Acesso em: 10 nov. 2004

Comunicado Técnico, 65

Ministério da Agricultura, Pecuária e Abastecimento

Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)
Endereço: Caixa Postal 6041 - Barão Geraldo
13083-970 - Campinas, SP
Fone: (19) 3789-5743
Fax: (19) 3289-9594
e-mail: sac@cnptia.embrapa.com.br

1ª edição on-line - 2005