Plesk: Plesk & Qmail: Spamprotection mit Greylisting

Was macht greylist?

Greylisting ist eine effektive Methode zur Spam-Protection. Es block jede Email beim ersten Versuch. Erst wenn diese Email ein zweites Mal eingeliefert werden soll, wird sie angenommen. Warum das funktioniert? Weil jeder echte Email-Server mehrmals versucht seine Emails bei dem Empfänger abzuliefern. Schließlich könnte ja mal ein Server ausgefallen oder überlastet sein. Spam-Versender-Programme hingegen arbeiten anders: Sie liefern einfach nur ein, ohne die Antwort zu registrieren. Es versucht also abgelehnte Emails i.d.R. kein zweites Mal einzuliefern. Und deshalb ist Greylisting so effektiv.

Das eigentliche Problem liegt daran, daß Greylisting sich in die Abarbeitung der Emails bereits beim Einliefern (SMTPd) einklinken muß. Bei Programmen wie Postfix ist dies recht einfach. Qmail ist etwas schwieriger, da die qmail-smtpd ersetzt werden muß. Bei einer std. Installation von Qmail wäre dies kein Problem, aber im Zusammenhang mit Plesk schon, da SWsoft eine spezielle Version von Qmail mit Plesk ausliefert.

Lösung:

(Aus dem SWsoft-Forum: qmail & greylisting spam control bzw. Adding Greylisting support to qmail on Plesk 8)

Installation
Vorbereitung der MySQL-Datenbank:

Mit `mysql` oder `phpMyAdmin` folgende User, Datenbank und Tabelle anlegen:

```
# Datenbank 'qmail':
CREATE DATABASE qmail;
# User 'greylist' (ändert evtl. das Passwort):
GRANT ALL ON qmail.* TO 'greylist'@'localhost' IDENTIFIED BY 'passwort';
FLUSH PRIVILEGES;
# Tabelle 'relaytofrom':
USE qmail;
CREATE TABLE relaytofrom (
  id bigint(20) NOT NULL auto_increment,
  relay_ip varchar(16) default NULL,
  mail_from varchar(255) default NULL,
  rcpt_to varchar(255) default NULL,
  block_expires datetime NOT NULL default '0000-00-00 00:00:00',
  record_expires datetime NOT NULL default '0000-00-00 00:00:00',
  blocked_count bigint(20) NOT NULL default '0',
  passed_count bigint(20) NOT NULL default '0',
  aborted_count bigint(20) NOT NULL default '0',
  origin_type enum('MANUAL','AUTO') NOT NULL default 'MANUAL',
  create_time datetime NOT NULL default '0000-00-00 00:00:00',
```

```
  last_update timestamp(14) NOT NULL,
  PRIMARY KEY  (id),
  KEY relay_ip (relay_ip),
  KEY mail_from (mail_from(20)),
  KEY rcpt_to (rcpt_to(20))
) TYPE=MyISAM;
```

Sourcen laden und bearbeiten

Zum Kompilieren werden folgende Pakete benötigt: `mysql-devel` und `openssl-devel`

```
#SuSE:
yast -i mysql-devel openssl-devel zlib-devel
#Debian:
apt-get install libmysqlclient12-dev libssl-dev
```

Download der Greylist-Sourcen:

(Diese Sourcen entahlten bereits die Patches von SWsoft.)

PS: Da Meshier den direkten Download unterbindet stelle ich die Pakete unter huschi.net bereit.)

```
cd /usr/local/src/
#für Plesk 7.5:
wget http://www.huschi.net/download/qmail-1.03-psa-published-greylist.tar.gz
#für Plesk 8:
wget http://www.huschi.net/download/qmail-103-greylist-psa8.tar.gz

#auspacken
tar xfz qmail-1*
cd qmail-1.03

#bei Plesk 8 fehlt leider eine Datei:
echo "-lssl -lcrypto" ›ssl.lib
```

Nun müssen die o.g. Datenbank-Zugangsdaten in die Datei `local_scan.c` eingetragen werden:

```
#define MYSQLHOST "localhost"
#define MYSQLUSER "greylist"
#define MYSQLPASS "passwort"
#define MYSQLDB   "qmail"
```

```
#define BLOCK_EXPIRE  4   /* minutes until email is accepted */
#define RECORD_EXPIRE 1500  /* minutes until record expires */
#define RECORD_EXPIRE_GOOD  36 /* days until record expires after accepting email */
```

Distributionsabhängige Änderungen:

Unter **Debian** muß man noch das `Makefile` anpassen:

```
edit Makefile
#ersetzte alle Vorkommen von "/usr/lib/mysql/libmysqlclient.a"
#durch "/usr/lib/libmysqlclient.a"
```

Auch bei einem **64bit-SuSE** müssen Anpassungen im `Makefile` vorgenommen werden:

```
edit Makefile
#ersetzte alle Vorkommen von "/usr/lib/mysql/libmysqlclient.a"
#durch "/usr/lib64/mysql/libmysqlclient.a"
```

Wer **Plesk 7.5** hat und in Haggybear's Plesk Greylisting Config Panel die Domain-Whitelist-Funktion nutzen will, muß folgenden Bugfix in die Datei `local_scan.c` einbauen:

```
#Suche nach "checkWhiteListDomain"
#Suche nun nach "sprintf(sql" (ca. 11 Zeilen weiter)
#gehe ans Ende der Zeile und ersetze:
rcpt_to = '%s'
#durch
rcpt_to LIKE '%%%s'
```

Greylist kompilieren

Nun geht es ans kompilieren:

```
make
```

ACHTUNG! Auf Fehlermeldungen achten und entsprechend reagieren!

Z.B. folgende

```
tls.c:12: error: conflicting types for 'strerror'
/usr/include/string.h:256: error: previous declaration of 'strerror' was here"
```

Wird gelöst indem man in `tls.c` Zeile 12 auskommentiert.

Greylist installieren

Die Installation besteht aus den Schritten: Qmail stoppen, nötige Dateien drüber kopieren, Qmail starten und Aufräum-Script installieren.

```
# Email-Empfang stoppen:
/etc/init.d/inetd stop
# oder:
/etc/init.d/xinetd stop
#(Qmail selbst muß nicht beendet werden.)

# installieren:
cp -p /var/qmail/bin/qmail-smtpd /var/qmail/bin/qmail-smtpd.old
cp qmail-envelope-scanner /var/qmail/bin/.
chown root.qmail /var/qmail/bin/qmail-envelope-scanner
cp -pf qmail-smtpd /var/qmail/bin/.
chown root.qmail /var/qmail/bin/qmail-smtpd

# qmail wieder starten:
/etc/init.d/inetd start
# bzw.  /etc/init.d/xinetd start
```

Erste Ergebnisse kann man mit `tail -f /tmp/greylist_dbg.txt` verfolgen. Dort sollte sowas drin stehen. (Falls nicht, weiter unten weiter lesen.)

```
protocol = notneeded4qmail tiqprotagofuj@protago.com
tiqprotagofuj@protago.com -› mail@meine-domain.de (88.235.162.111) Doesn't Exists Block
```

Aufräum-Script

Damit die Datenbank und das Logfile nicht überlaufen, schreiben wir ein Perl-Script (Download unter Software für Qmail) nach `/etc/cron.daily/qmail-greylist-cleanup.pl`:

```
#!/usr/bin/perl -w
use strict;
```

```
use constant DBD => 'DBI:mysql:qmail:localhost:3306';
use constant DBUSER => 'greylist';
use constant DBPASS => 'passwort';

use DBI;

system ('cat /dev/null > /tmp/greylist_dbg.txt');

my $dbh = DBI->connect(DBD,DBUSER,DBPASS) or die "can't connect to db ", $DBI::errstr,
":$!";

$dbh->do("DELETE FROM relaytofrom WHERE record_expires < NOW() - INTERVAL 1 HOUR AND
origin_type = 'AUTO'");
$dbh->do("OPTIMIZE TABLE relaytofrom");

$dbh->disconnect;
```

Nicht vergessen: Ausführbar machen:

```
chmod +x /etc/cron.daily/qmail-greylist-cleanup.pl
```

Probleme / Debugging

Wie immer gilt der Grundsatz: Logfiles!

Gerade das Debug-Log von Greylisting ist wichtig: `/tmp/greylist_dbg.txt`

Aber auch das allgemeine Maillog: `/var/log/mail.info`

Logging von MySQL-Fehlern:

Ein häufiges Problem ist, daß `qmail-envelope-scanner` sich nicht in die Datenbank einloggen kann. Hier hilft nur ganz genau nochmal alle eingetragenen Passwörter zu prüfen.

Falls das immer noch nicht hilft muß eine kleines Debugging in die `local_scan.c` eingebaut werden: (ca. Zeile 284)

```
// if(mysql) mysql_close(mysql);
 if (dbglog) {
   if (mysql) {
       if (mysql_errno(mysql))
           fprintf(dbglog, "MySQL-Error: %s
", mysql_error(mysql));
```

```
        mysql_close(mysql);
    } else {
        fprintf(dbglog, "MySQL-Error: %s
", "cannot initialize!");
    }
    fprintf(dbglog, "--------
");
    fclose(dbglog);
 }
```

Danach nochmal `make` durchlaufen lassen und wieder die Programm-Datei `qmail-envelope-scanner` nach `/var/qmail/bin/` kopieren.

Nun sollte der Grund für die Login-Probleme in der Datei `/tmp/greylist_dbg.txt` ersichtlich sein.

Problem mit TLS

Ein weiteres Problem ist in Verbindung mit TLS aufgetaucht. Bitte testet `qmail-smtpd` nach der Installation mit openssl ob der TLS-Handshake funktioniert:

```
openssl s_client -crlf -starttls smtp -connect localhost:25
```

Bricht er direkt nach 2 oder 3 Zeilen ab, so war er nicht erfolgreich. Bringt er hingegen das ganze TLS-Zertifikat und noch einige Debug-Informationen dazu, so muß man den Connect mit `quit` beenden.

Wenn es nicht erfolgreich war, muß eine Cipher-List erstellt werden:

```
openssl ciphers > /var/qmail/control/tlsclientciphers
openssl ciphers > /var/qmail/control/tlsserverciphers
```

Direkt im Anschluß wieder den TLS-Handshake mit o.g. Befehl testen.

Eindeutige ID: #1124
huschi
2008-07-28 20:39