

EHLO

März 2010

Wolfgang Breyha



SPAM Workshop Teil 2

Themenüberblick

- Spamtraps
- DCC/Razor/Pyzor
- ClamAV/Saneseconomy
- SpamAssassin (Config, Modules, Rules)
- DKIM
- Logfileanalyse
- Feedbackschleifen
- Maßnahmen gegen ausgehenden Spam
- exim Beispiele



Phishing

Betreff: ***ACCOUNT UPGRADE***
Datum: 19 Nov 2009 21:49:08 -0000
Von: UNIVERSITY OF VIENNA <support@univie.ac.at>
Antwort an: s.team43@yahoo.com
An:@univie.ac.at

----Copyright © 2009. UNIVIE.AC.AT-----

Account Subscriber,

Due to excess abandoned webmail accounts, We are currently performing maintenance on our Digital webmail Server to improve the spam filter services in our webmail systems for better online services to avoid virus and spam mails. In order to ensure you do not experience service interruption, respond to this email immediately and enter your Username/id here (*****) password here (*****) and future password here (*****). Checkout new features and enhancements with our newly improved and secured webmail.

NB: We require your username and password for Identification purpose only.

----Copyright © 2009. UNIVIE.AC.AT-----

Zitat User: sieht ziemlich echt aus!

März 2010

Wolfgang Breyha



Phishing

Lieber univie.ac.at Subscriber,

Diese Nachricht ist vom univie.ac.at Nachrichtenübermittlungszentrum bis
alle Benutzer dieses Gebiets.

Wir befördern zurzeit unsere Datenbasis und schicken Kontozentrum per Email.

Wir löschen die ganze unbenutzte E-Mail-Rechnung, um mehr Raum auf neue
Rechnungen zu schaffen. Um Ihre Rechnung davon abzuhalten, zu schließen
werden Sie es unten aktualisieren müssen, so dass wir wissen werden, dass
das zurzeit eine verwendete Rechnung ist.

BESTÄTIGEN SIE IHRE E-MAIL-IDENTITÄDT UNTER der E-Mail

Benutzername:

Kennwort:

Datum von Birth:

Land oder Territorium:

.....

März 2010

Wolfgang Breyha



Spamtraps

- komplette Domains oder Hostnames
- unknown users
- Webseiten mit versteckten Mailadressen für spider
- “unsubscribe” Links
- usenet

- im Zweifelsfall fake reject



DCC – Distributed Checksum Clearinghouses

- <http://www.rhyolite.com/dcc/>
- 3 Hashwerte unterschiedlicher Abstraktionsstufen
 - body, fuz1 und fuz2
 - getrennt gezählt und als Ergebnis geliefert
- erlaubt höhere Wertung von Spamtraps
- greylisting mit body hash oder klassischem tripple aus from, rcpt und ip
- exim ACL patch
<http://www.blafasel.at/exim-dccacl/>
seit 4.70 inkludiert



Pyzor

- <http://sourceforge.net/apps/trac/pyzor/>
 - in den Anfängen ähnlich wie Razor
 - GPL
 - berechnet einen Wert nach definierten Regeln
- <http://sourceforge.net/apps/trac/pyzor/wiki/About>



Razor2/Cloudmark

- <http://razor.sourceforge.net/>
- open source Variante von kommerziellem Cloudmark Authority
- Server werden von Cloudmark betrieben
 - Registrierung notwendig
- 4 verschieden Filter Engines



ClamAV/saneseconomy

- <http://www.saneseconomy.com/clamav/>
- mehrerer DBs für SPAM und Phishing Signaturen
- eigenes update Skript
- Signaturen überschneiden sich mit offiziellem ClamAV
 - eigene Instanz notwendig
- hohe Trefferquote
 - 5.10.2009: 22165 Mails > 8 Punkte,
5324 saneseconomy Treffer
- Anbindung zB. über ClamAV Plugin



SpamAssassin

- Unmengen lokaler und Netzwerktests
 - regex
 - DNSBLs (IP, URI, Sender)
 - SPF, DKIM
 - Image Metadaten (ImageInfo)
 - Spracherkennung (textcat)
 - DCC, razor2, pyzor
 - SpamCop reporter
 - RelayCountry



SpamAssassin

- Auf Userbasis
 - Auto Whitelist
 - Bayes
- Viele zusätzliche Plugins und Rulesets



user based checks

- schwierig auf MTAs zu implementieren
 - DB Backend notwendig wenn keine lokalen User
- Problem im SMTP Dialog: 1 Mail, viele Empfänger
 - allg. dummy User verwenden
- auf großen Systemen besser an den MUA abgeben
- möglichst viele Header setzen die Bayes Tokens liefern



mitgelieferte rules und plugins

- nach Möglichkeit viele Plugins aktivieren (.pre Dateien)
- nur im Ernstfall (bei false positives) an den scores schrauben
- sa-update verwenden!
- Rule2XSBody + sa-compile verwenden



Addons (rulesets)

- <http://wiki.apache.org/spamassassin/CustomRulesets>
- <http://www.rulesemporium.com/> (SARE)
- http://zmi.at/x/70_zmi_german.cf



Addons (plugins)

- <http://wiki.apache.org/spamassassin/CustomPlugins>
- iXhash
- Botnet
- FuzzyOCR
- p0f-analyzer



eigene rules

- zusätzliche RBL (zB iX)

```
ifplugin Mail::SpamAssassin::Plugin::URIDNSBL
```

```
header RCVD_IN_NIXSPAM
```

```
eval:check_rbl('nixspam',  
               'ix.dnsbl.manitu.net.')
```

```
describe RCVD_IN_NIXSPAM
```

```
Received via a relay in NiX-SPAM
```

```
tflags RCVD_IN_NIXSPAM
```

```
net
```

```
score RCVD_IN_NIXSPAM
```

```
1.0
```

```
endif # Mail::SpamAssassin::Plugin::URIDNSBL
```



eigene rules

```
header __ZIDWEBMAILREPLYTO      exists:Reply-To
body   __ZIDWEBMAILPASSWORD     / (kenn|pass)wor(d|t)/i

meta   __ZIDPHISHCOUNT         ( __ZIDWEBMAILSUBJECT +
                               __ZIDWEBMAILREPLYTO ... )

meta   ZIDWEBMAILPHISHLOW       __ZIDPHISHCOUNT > 3
describe ZIDWEBMAILPHISHLOW    maybe a webmail phishing attempt
score   ZIDWEBMAILPHISHLOW     2
```



Performance

- Darauf achten wann Komponenten Mails via Filesystem an Scanner weiterreichen müssen.
- RAM Disk verwenden
- Ergebnisse vom MTA via Header weiterreichen
 - spart I/O und doppelte checks
 - simple regex im SpamAssassin
- sa-compile
- caching DNS (unbound, bind)



DKIM

DomainKeys Identified Mail

- <http://www.dkim.org/>
- gmane.ietf.dkim
- Erweiterte Kombination aus
 - Yahoo! DomainKeys
 - CISCO Identified Mail
- Erster Baustein im Mai 2007 => RFC 4871
- Author Domain Signing Practices (ADSP)
seit August 2009 => RFC 5617



DKIM technisches

- signiert Teile des Headers und den Body
- Signatur im Mailheader

```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;  
d=univie.ac.at; s=rev1; h=Message-ID:Date:From:MIME-Version:To:  
Subject:Content-Type:Content-Transfer-Encoding; bh=CHKp57xvG+TkL  
tX7hfa7jYenETIpLWpRR7c1cM4GJ3E=; b=bxS//cYqDJTBuZ93e2rmpZyyVmpHP
```

....

- **PublicKey** vorerst als DNS TXT

```
# host -t txt rev1._domainkey.univie.ac.at  
rev1._domainkey.univie.ac.at descriptive text "v=DKIM1; k=rsa; g=*; s=email; t=y; p=MIGfMA....."
```

- signiert bzw. verifiziert wird durch border MTAs.



DKIM vs. SPF

- bezieht sich auf From: Header anstatt auf envelope from
- keine TXT Records für die Domain selbst
 - `<selector>._domainkey.<domain>`
- keine Verifikation des Pfades
 - keine Probleme mit Forwards (SRS)
 - leichtere Sender Delegation
- 867641 mail from => 123904 Mails
SPF: 48582 pass; 1451 fail; 1405 softfail; 1286 neutral
DKIM: 25028 signed; 20853 verified



ADSP

- TXT RR `_adsp._domainkey.<domain>`
- keine Vererbung
- `dkim=(unknown|all|discardable)`
- Ergebnisse: none, pass, unknown, fail (all), discard



DKIM Probleme

- RFC 4871 definiert nur den technischen Rahmen
- SSP=>ASP=>ADSP (Author Domain Signing Practices) seit August 2009 RFC 5617. Jedoch kaum definiert
- Probleme mit Mailinglisten
 - Signierter From: Header
 - Veränderungen um Body (Footer, ...)
- fehlerhafte Implementierungen
- Admins wie bei SPF scheinbar überfordert



Derzeitiger Nutzen von DKIM

- Uni Wien signiert seit Jänner 2008 ausgehende Mails
- eingehende Mails mit univie.ac.at Domain werden teils auf gültige Signaturen geprüft um Phishing zu unterbinden. zB.: postmaster@univie.ac.at
- SpamAssassin ruleset
- <http://www.dkim-reputation.org/>
- echo@univie.ac.at verifiziert eingehende Mail und signiert die Antwort



Logfileanalyse

- Logging via syslog in textfiles
 - gzip bzw. bzip2 bei täglicher Rotation
- Tools zur Analyse
 - allg. GNU-Tools: grep, sed, awk, sort, uniq
 - spezielle Tools: exigrep, exitop
- realtime Grafiken erzeugen
 - syslog-ng => skript => counter
 - RRDs aus snmp oder textfiles



Feedback

- Vorstufe für manuelles feedback bauen
 - Bereitstellung vermutlich interessanter Werte via Webinterface
 - Wo sinnvoll Automatismus implementieren
- syslog-ng => Überwachung durch Skript
- Beispiele
 - Grenzwerte für bad recipients, greylisting und hits auf spamtraps. Triggert greylisting und blocks.
 - Grenzwerte für ausgehende Mails pro envelope Sender
 - MX retry check



rbldnsd

- feedback für MTA mittels lokaler RBL basierend auf rbldnsd
 - optimiert für RBLs
 - extrem einfacher Aufbau der Zonefiles

```
#$TTL 2048s
#$SOA 10m blacklist.univie.ac.at postmaster.univie.ac.at 1...
85.88.6.85:127.0.0.2:und aus
:127.0.0.2:you have won free email access! contact abuse@univie.ac.at \
to grep it!
80.253.80.0/24
```

- hoch performant
- antwortet während reload
- für IP und URI geeignet



Feedback

- gesperrte IPs/Hostnamen kein Problem
- gesperrte EMail-Adressen durch `s/\@/.x-at-x./`

```
#$TTL 30s
#$SOA 10m autouri.univie.ac.at postmaster.univie.ac.at 11...
:127.0.0.32:kryoknast
wbfreezetest.x-at-x.univie.ac.at :127.0.0.6:hold the line
```

- check im MTA mit URIBL lookup
 - zB. in exim mit

```
warn    set acl_m0      = ${quote_local_part: \
                $sender_address_local_part}.x-at-x.$sender_address_domain
dnslists = wanted.univie.ac.at=127.0.0.6/$acl_m0
log_message = FROZEN: $sender_address found on
                wanted.univie.ac.at
control   = freeze/no_tell
```



Fragen?

