Csak most, csak neked – Spamszűrők és a spam lélektana

Ma már senkinek nem kell elmagyarázni a kéretlen levél fogalmát. Ha valakinek publikus email címe van, az szinte biztosan kapott már szexuális ajánlatot elektronikus levélben, és mivel a spammerek a szükséges mennyiségű empátiaval is rendelkeznek, rögtön ajánlottak némi Viagrát is.

Valaki kiszámolta, hogy egy átlag felhasználó csak a spamek törlésével órákban mérhető időt veszít el évente. Ha már te is unod a ***Del*** gomb mindennapi kötelező ujjgyakorlatát, akkor a spamszűrők lehetnek segítségedre, amelyek a beérkező levél bizonyos jellemzői alapján, különféle módszerek alkalmazásával döntik el, hogy az spam (szemét; kéretlen levél) vagy ham (hasznos levél). Az alábbiakban különféle módszereket ismertetek a spam felismerésére. A spamszűrő alapvetően két helyen működhet: kliens- vagy szerveroldalon.

Mi a spam?

A spam olyan elektronikus levél, amelynek számunkra semmi értelme, és fölöslegesen foglalja erőforrásainkat (a gép és ember idejét), arról már nem is szólva, hogy ezek tartalma sok esetben félrevezető, pl. a nigériai milliókról sajnos többen bedőltek. Az előbb említett meghatározásból az is következik, hogy ami számomra kéretlen és bosszantó reklám (spam), az másnak esetleg értékes levél (ham). Ez pedig alaposan megnehezíti a spam szerveroldali szűrését.

Kulcsszó alapú szűrés

A spam szűrésének legegyszerűbb módja az, ha bizonyos szavakat keresünk a levélben (például ***szex***, ***VIAGRA***), és ha megtaláljuk a levél fejlécében vagy a törzsében, akkor azt spamnek tekintjük. A módszert gyakorlatilag minden alkalmazás támogatja mind kliens- (például ***Thunderbird***), mind pedig szerveroldalon (például ***procmail***, ***maildrop***, ***postfix***). Azonban az egyszerűsége egyúttal hátrány is. A spammerek ugyanis úgy kerülték meg ezeket a szűrőket, hogy például a ***VIAGRA*** szóból lett ***V1AGRA***, ***V.I.A.G.R.A***, stb, amit a szűrő már nem azonosított spamként – de mi még igen. A variációk száma pedig olyan nagy, hogy egy kulcsszó listát manuálisan karbantartani képtelenség a gyakorlatban. További hátrány, hogy ha így akarunk pl. a szex hirdetésektől megszabadulni, akkor az olyan levél is a kukában végzi, amely pl. a szextáns szót tartalmazza és nem spam, arról nem is beszélve, hogy a ***szex*** szó legitim levelekben is előfordulhat.

A kulcsszó alapú módszer változata a reguláris kifejezésre történő szűrés, amivel összetettebb és hatékonyabb szűrés végezhető. Ez a módszert – korlátozottan – akár férgek szűrésére is használható, ha szerepel azokban egy ismétlődő minta.

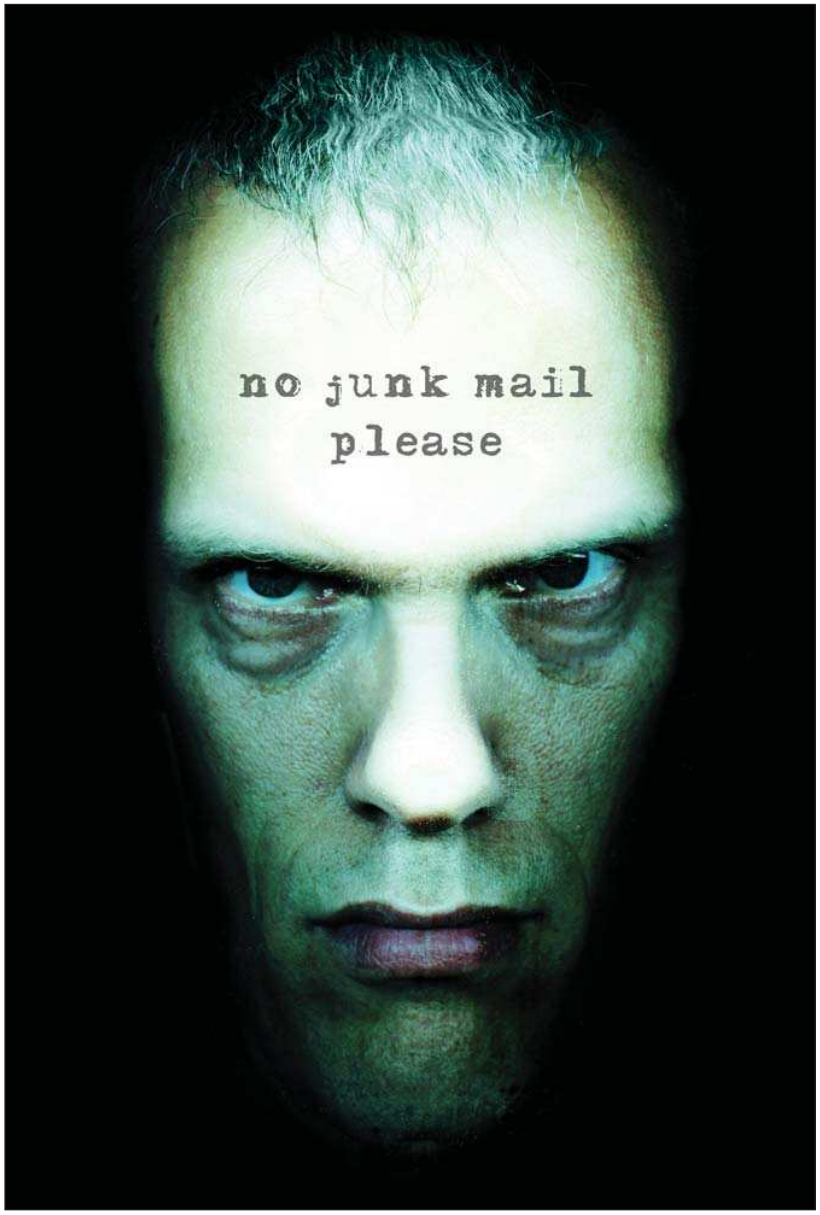
Feketelista

Ez az eljárás a levél fejlécében szereplő ***IP*** címeket veti össze különféle ***RBL*** listák, például ***MAPS*** (➲ ***http://www.mail-abuse.com/***),

ORDB (➲ *http://www.ordb.org/*) adatbázisával, és megvizsgálja, hogy azok szerint a minket megszólító gép már ismert spammer vagy levéltovábbító (***open relay***). (Ilyen lehet például egy megfertőzött zombi.) A módszer a gyakorlatban egy egyszerű *DNS* lekérdezést jelent, és a kapott válasz alapján lehet eldönteni, hogy az ***RBL lista szerint*** az adott gép spam forrás vagy sem. A feketelisták úgynevezett fehérlistákkal (whitelist) is kombinálhatók, amely listán szereplő *IP* címeket mindig átenged a szűrő. Ez a technika egyszerű, viszonylag kevés erőforrást használ, széles körben implementált, és szinte minden modern levelezőkliens és -szerver támogatja. Természetesen nem csak a levél fejlécében szereplő címeket, neveket lehet feketelistákkal összevetni, de akár a levél törzsében levőket is, sőt az *SMTP* kapcsolat során megadott neveket, címeket is (pl. a *HELO*, *MAIL FROM*, *RCPT TO* utáni paramétereket), illetve a kliens *IP* címét is, és ha annak nincs korrekt *PTR* rekordja, az minimum gyanús.

Nem mindenki elégedett azonban a feketelisták hatékonyságával. Előfordul, hogy egyes *RBL* listákon egy teljes C-osztályt tiltanak ki akkor is, ha csak egyetlen renitens küld spamet az adott hálózatból. Ez különösen a szerverbérlők (colocation szolgáltatás) számára lehet igen kellemetlen, mert így ártatlanul is tiltólistára kerülhetnek, a helyzet megoldása pedig sokszor nem kevés utánajárást igényel. A spammerek pedig igen ritkán használják saját azonosítható gépüket kéretlen levelek küldésére, inkább feltört zombikat vagy korlátozás nélküli továbbítószervereket (***open relay***) használnak erre. Ezekből meg éppen elég van ahhoz, hogy ne nagyon viselje meg őket, ha pár darab tiltólistára kerül. Paul *Graham* a *„Filters vs Blacklists”* című írásában (➲ *http://www.paulgraham.com/falsepositives.html*) rámutat, hogy például a *MAPS* feketelista úgy fogta meg a spamek 24%-t, hogy közben az ártatlan levelek 34%-t spamnek tekintette (false positives).

A feketelista egyik változatában nem konzultálunk *RBL* listákkal, hanem a tiltást *IP* szinten végezzük, mondjuk egy csomagszűrő alkalmazás (például *ipf*, *iptables*) segítségével, esetleg az adott



IP címet a ***loopback*** interfész felé irányítjuk. Ebben az esetben azonban nekünk kell kézzel – vagy más módon – karbantartanunk az ehhez szükséges ***netfilter*** illetve ***route*** bejegyzéseket.

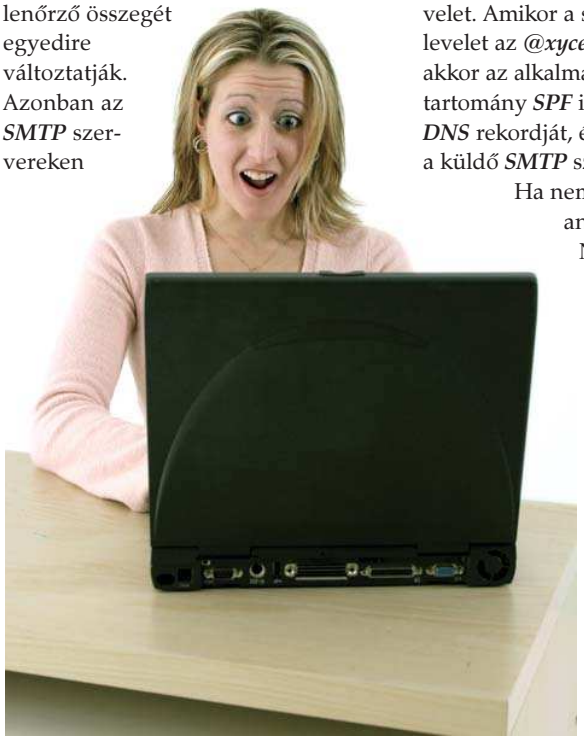
Szürkelista (greylist)

A szürkelista a spammereknek azt a szokását használja ki, hogy azok az első sikertelen próbálkozás után általában feladják, és továbblépnek a következő címzettre. Amikor az *SMTP* kiszolgálónk levelet kap, azt először 450-es hibaüzenettel (átmeneti hiba) elutasítja. A szabványoknak megfelelően működő *SMTP* szerverek egy bizonyos idő múlva újra próbálkoznak. A mi szerverünk azonban feljegyezte a küldő *SMTP* szerver azonosítóját, a feladó és a címzett email címét, továbbá az időt (egy bizonyos idő múlva ez a bejegyzés lejár), és a küldő *SMTP* szerver második próbálkozását már elfogadja. Így a spammerek többsége nem jut be, míg a legitim levelezőpartnerek igen. Ezt a megoldást támogatja például a *Postfix* egy egyszerű ***policy démon*** segítségével (➲ *http://www.postfix.org/addon.html*). A *Postfix* levelező listán többen igen kedvező tapasztalatokról számoltak be, míg néhányan kifogásolták a levelek késleltetését. Itt is lehetőség van fehérlista definiálására, ahol meg lehet adni azokat az *IP* címeket/feladókat/címzetteket, amelyeket azonnal

elfogad a szerver. Jelenleg ez a módszer jó hatásfokkal dolgozik, de ez könnyen a múlté lehet, ha a spammerek alkalmazkodnak hozzá, és újraküldik a spamet. Ehhez azonban nagyobb teljesítményű infrastruktúrára van szükségük, ami nekik több pénzbe kerül, és ez nekünk – akik már torkig vagyunk a spammel – jó.

Distributed Checksum Clearinghouse (DCC)

A spameket általában nagy tömegben küldik ki, nem ritka, hogy némelyik több milliós példányszámot is elér. A *DCC* lényege az, hogy az ebben résztvevő minden *SMTP* szerver minden egyes beérkező levélből egy ellenőrzőösszeget képez, amit elküld a *DCC* szervereknek. A *DCC* szerverek nyilvántartják, hogy az egyes ellenőrző összegekhez hány találat tartozik. Az *SMTP* szerverek az elküldött ellenőrző összegre kapnak egy választ (egy számot). Ha ez a szám meghalad egy értéket, akkor nyilvánvalóan tömegesen kiküldött levélről, azaz spamről van szó.
A spammerek azonban ezt az eljárást is igyekeznek megkerülni (a spam fejlődik) úgy, hogy az egyes levelekbe véletlenszerű tartalmat tesznek, így kvázi testre szabják a leveleket (amihez egyetlen szó is elég), ezzel minden egyes levél ellenőrző összegét egyedire változtatják. Azonban az *SMTP* szervereken futó *DCC* kliens is úgy módosult, hogy többféle ellenőrző összeget képez, amivel képes kiküszöbölni, hogy a csupán 1-2 apró részletben (pl. idő, véletlenszerű azonosító) eltérő spamek különböző ellenőrzőösszeget (*checksum*) eredményezzenek.
A módszer annál hatékonyabb, minél többen vesznek benne részt. Eredményesen képes felismerni a nagy tömegben kiküldött spamet. Hátránya, hogy külön alkalmazás szükséges hozzá, és viszonylag bonyolult együttműködésre bírni a levelező alkalmazásunkkal. Ilyen alkalmazás pl. a *DCC* (➲ *http://www.rhyolite.com/anti-spam/dcc/*) és a *Vipul's Razor* (➲ *http://razor.sourceforge.net/*)



Sender Policy Framework (SPF)

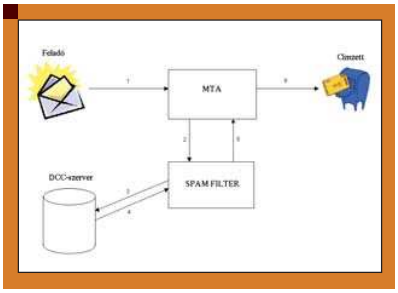
Megnehezíti a spammerek felderítését, hogy meghamisítják a levél fejlécében szereplő email címeket. Semeddig nem tart egy olyan levelet készíteni, amelyik pont úgy néz ki, mintha az X cég vezérigazgatója küldte volna. Mennyivel könnyebb lenne az életünk, ha biztosak lehetnénk abban, hogy egy levél feladója garantáltan az *@xyceg.hu*. Pont erre találták ki az *SPF*-et. Az adott tartományban el kell helyezni egy rekordot (jelenleg a tartomány *TXT* rekordja), amely definiálja, hogy az adott tartomány nevében mely *SMTP* szerverek küldhetnek levelet. Amikor a szerverünk kap egy levelet az *@xyceg.hu* tartományból, akkor az alkalmazás lekérdezi az adott tartomány *SPF* információt tartalmazó *DNS* rekordját, és ellenőrzi, hogy a küldő *SMTP* szerver szerepel-e ott. Ha nem, akkor nyilvánvalóan spamről van szó.
Noha a módszer működik, még nem terjedt el széles körben, így az *SPF*-et támogató *SMTP* szervernek egyelőre el kell fogadnia azokat a leveleket, amelyekben a feladó tartománya nem tartalmaz *SPF* információt. Továbbá vannak még meg nem oldott kérdések ez ügyben, pl. levelező listák kezelése.
Az *SPF* honlapján (➲ *http://spf.pobox.com/*) varázsló segíti a megfelelő *DNS* rekord elkészítését, amit érdemes publikálni, hogy minél kevesebb esélyt adjunk arra, hogy a spammerek a mi tartományunkat hamisításra használják fel.

Kérdés-válasz (challenge-response) szűrők

Amikor a szerver egy ismeretlen feladótól kap egy levelet, akkor automatikusan visszaküld egy kérdést (challenge) tartalmazó üzenetet. Ez lehet csupán annyi, hogy „válaszoljon erre a levélre", de igényelheti egy komplex web űrlap kitöltését is. Ha a kért műveletet végrehajtja a feladó, akkor a rendszer elfogadja az emailt. Mivel a spammerek általában hamisított feladó email címet használnak, ez a megoldás minden ilyen levelet blokkol. Ha pedig mégis az ő saját, élő címüket használják, akkor sem igen létezik olyan tömeges levélküldő alkalmazás (*bulk mailer*), amely nagy mennyiségben lenne képes ezekre a kérdésekre válaszolni.
Noha ez a módszer hatékony a spammerek ellen, van azonban néhány kellemetlen mellékhatása is, pl. kérdéses, hogy levelező listák esetében mennyire járható ez az út, és ez érvényes minden olyan levélre, amit nem ember adott fel (például űrlap kitöltésének e-mailben történő automatikus visszaigazolása).
A módszer egyik változata egy olyan speciális „bélyeg" meglétét követeli meg a levélben, amelynek elkészítése relatíve számításigényes. A legitim alkalmazások számára ez nem okoz számottevő terhelést, de tömeges levélküldésnél már olyan sok gépidőre van szükség a bélyegek előállítására, hogy a *spamware* program teljesítménye (a kiküldött levelek száma) erősen lecsökken.

Bayesian szűrők

Bár az előbb ismertetett módszerek több-kevesebb hatékonysággal működnek, van egy még jobb megoldás, mégpedig a levelek tartalmának elemzése. Az eljárás *Thomas Bayes* matematikus után kapta a nevét, aki egy valószínűségelmélettel kapcsolatos képletet dolgozott ki. A módszer lényege az, hogy amikor egy levelet kapunk, a *Bayesian* program azt szavakra (precízebben szólva tokenekre)



bontja. Két adatbázis (*HAM* és *SPAM*) segítségével minden egyes tokenhez meghatároz egy valószínűséget, mekkora az esélye, hogy a levél spam, ha az adott token szerepel benne. Ezután a tokeneket sorrendbe állítja aszerint, hogy valószínűségük mennyire tér el egy semleges középértéktől (amiről nem tudjuk eldönteni, hogy ham vagy spam). Végül veszi a középértéktől leginkább eltérő N darabot, és egy összetett képlet segítségével kiszámítja azok összesített spam valószínűségét. Az alkalmazás általában a levél fejlécébe beszúr egy extra mezőt, amiből kiderül, hogy az adott levél szerinte spam vagy sem, így el tudjuk dönteni, mit tegyünk ezek után a levéllel.
A *Bayesian* szűrőt azonban használat előtt tanítani kell, azaz létre kell hoznunk a spam illetve a ham adatbázist, amelyhez a leveleknek egy olyan halmazát kell képezni (például egy mailbox spoolfile), amelyben csak spam illetve csak ham van.
Ez a szűrés is végezhető a felhasználó oldalán, például a *Thunderbird* beépített *Bayesian* szűrővel rendelkezik, és már viszonylag kevés (<100) üzenetnél is jól osztályozza a beérkező leveleket. Több alkalmazás létezik, amelyik szerveroldalon valósítja meg ezt a funkciót. A *bogofilter* parancssorból futtatható, így egyszerűen használható például *maildroppal* együtt a levelek osztályozására. Szerveroldalon is számos alkalmazás támogatja a *Bayesian* algoritmust, ill. annak különböző módosított változatait. Ezen alkalmazások közül némelyik *SMTP* protokollon kommunikál az *MTA*-val. A *Postfix* ezekkel is képes együttműködni.
A módszer előnye, hogy a téves pozitív azonosítások száma minimális, véleményem szerint a legjobb megoldás a felsorolt eljárások közül. Hátránya, hogy az összes közül a legerőforrásigényesebb, továbbá tanítani kell, mielőtt használható.
Ha valaki többet szeretne megtudni a *Bayesian* algoritmus lelkivilágáról, ajánlom figyelmébe a korábban említett *Paul Graham* honlapja (➲ *http://www.paulgraham.com/*) mellett *Gary Robinson* két írását: *„Gary Robinson's Rants"* (➲ *http://radio.weblogs.com/0101454/stories/2002/09/16/spamDetection.html*) és *„A Statistical Approach to the Spam Problem"* (➲ *http://www.linuxjournal.com/article/6467*)

Heurisztikus szűrők

Ez a módszer azt használja ki, hogy a spamek szokatlan jellegzetességekkel rendelkeznek. Ezért egy sor tesztet végez el a leveleken, és minden egyes szokatlan, spamre jellemző dolgot pontoz. Ha a pontok összege elér egy határt, akkor a levelet spamnek tekintjük. Büntetőpont jár azért, ha pl. hiányzik a `From:` vagy `To:` mező; ha ezek tartalma érvénytelen; ha a címzett nem szerepel sem a `To:` sem pedig a `Cc:` mezőkben; ha a `From:` és a `To:` mezők azonosak; ha több, mint 10 cím szerepel a `To:` vagy `Cc:` mezőkben; ha hiányzik vagy érvénytelen a `Message ID:`; ha a levél csak *HTML* részt tartalmaz, közönséges szöveget nem. Bizonyos levelekben csak egy – azonosítót is tartalmazó – link szerepel, amire kattintva lehet elolvasni a spamet. A cikk írása napján lettem figyelmes egy ➲ *www.xxxx.spammerdomain.com* alakú linkre, ahol az *xxxx* egy több karakterből álló egyedi azonosító, azaz egy wildcard tartománynévről van szó. Egy másik változatban valamelyik ismert keresőgép felparaméterezett *URL*-je szerepel, amelyre kattintva jelenik meg a spammer oldala. Érdemes ezeket is pontokkal „jutalmazni". A *SpamAssassin* (➲ *http://spamassassin.apache.org/*) ilyen típusú szűrésre (is) képes.

Egyéb szűrési lehetőségek

Előfordul, hogy csak egyetlen csatolt képet kapunk, amely tartalmazza a kéretlen üzenetet. Ez ellen például képfelismerő (*OCR*) alkalmazásokkal lehet védekezni, amelyek képesek kinyerni a szöveget a képből, aztán jöhet az elemzés.
A *Bayesian* elv egyik továbbfejlesztett változata a szavak kombinációját is figyeli, amely még jobb eredményt adhat. Ilyen például a *CRM114* nevű diszkriminátor (➲ *http://crm114.sourceforge.net/*).
A feketelista egy további változata esetén a gyanús *IP* címekre sebességkorlátozást (*traffic shaping*) lehet végezni. Ebben az esetben a vélt spammerek eleve csak korlátozott mennyiségű adatot vihetnek át időegység alatt. Ennek egy másik változatában az *SMTP* kiszolgálónk az ismeretlen kliensekkel szándékosan lassan kommunikál, pl. több másodpercet is vár, amíg egyáltalán `220 SMTP` bannert ad. Ez az ötlet arra épít, hogy a spammerek türelmetlenek, minél rövidebb idő alatt minél több levelet akarnak elküldeni.
Ha szerencsénk van, a spammer program leidőzít, és odébb áll.

Melyiket használjam?

E sorok írója szerint a legjobb választás, ha kombináljuk a fentebb ismertetett módszereket, ill. mind szerver-, mind pedig kliensoldalon alkalmazunk spamszűrést. Szerveroldalon a *Bayesian* szűrést javaslom egyesíteni a szürkelistával és a heurisztikus szűréssel, míg kliensoldalon jó döntés a *Thunderbird* adaptív spamszűrője. De a kulcsszó szerinti szűrés is jól használható, ha pl. keleti spammel kell megbirkóznunk. Azt is figyelembe kell venni, hogy a spam fejlődik, így az a technika, amely ma hatásos ellene, az holnap már lehet, hogy nem működik. Minél többféle szűrőt használunk, annál biztosabban tudjuk eldönteni egy levélről, hogy az spam vagy sem. De annál több erőforrást is igényel.
Egy jól működő spamszűrő kombináció esetén felmerülhet az az igény is, hogy immár ne egy kijelölt folderbe gyűjtsük a spamet, hanem egyszerűen dobjuk el, a felhasználókhoz már meg se érkezzen. Azonban úgy vélem, hogy ez nem jó ötlet, mert időnként mindegyik szűrő hibázik, és egy elvesztett levél akár sok elvesztett pénzt is jelenthet.

Sütő János
(jsuto@freemail.hu)
1997 óta használ Slackware Linux-ot. Szabadidejében a postfix clapf nevű vírus- és spamszűrőjét polírozza.