Detecting Spam and Spam Responses

FloCon 2009
Timothy J. Shimeall, Ph.D.

CERT/Network Situational Awareness Group

Overview

Why worry about email

Spam

Spam Responses

Why worry about E-mail?

Email: at the intersection of ‘business essential’ and ‘favored point of attack’

Well established usage

Well established weaknesses

Track record of exploitation

Spam and Spam Sources

Spam is unwanted, fraudulent, or malicious electronic mail

Spam source is a host that sends spam

At the protocol level, spam is correctly formed and sent

- Proper port
- Proper messages
- Complete TCP session

Spam Blockers Are Not Enough

Want analytical tool

Blocking message by message is costly and unreliable

- Treats symptoms, not problem
- Increasing fraction of traffic
- Difficult to automatically recognize
- Content recognition frequently evaded

Blocking fixed list of addresses is uncertain

- Frequently evaded
- Uncertain add/drop conditions
- Uncertain period of coverage
- Arms race

So How are Spam Sources Different?

Rate of sending email (well known)

- Spam is effective only when acted on (open attachment or link, reply, follow advice, etc.)
- Only some users will act on spam
- The more spam sent, the more will be effective

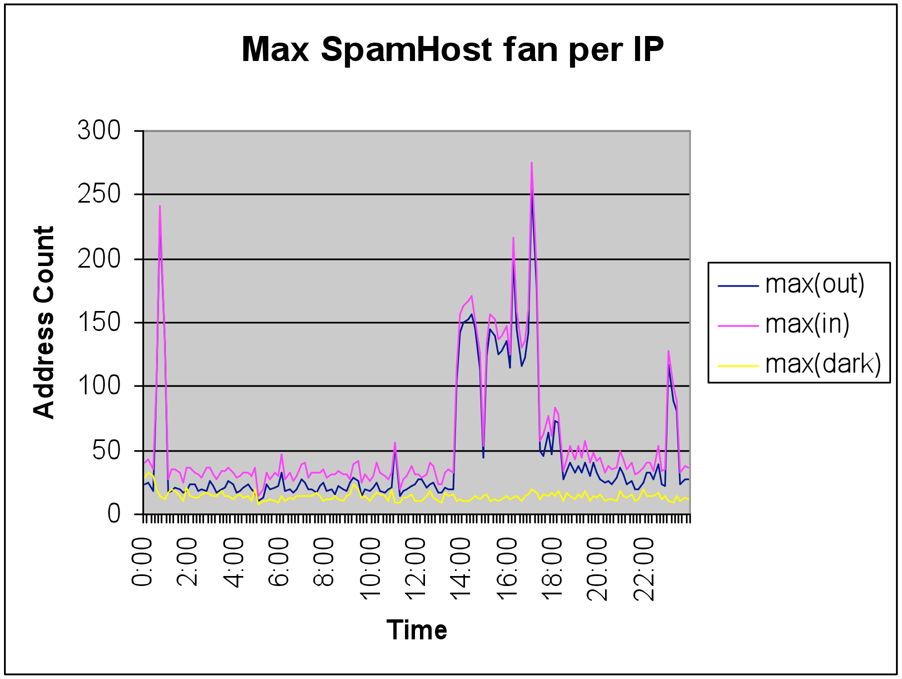
Locality of sending email

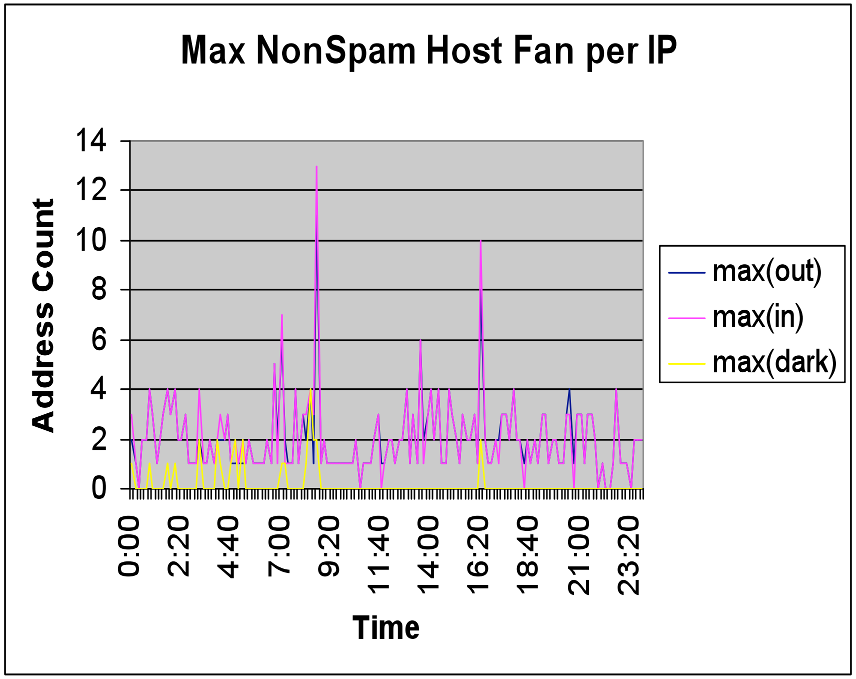
- Diversify the target
- Avoid collateral protection at target
- Normal email tends to be localized (most of your desired email comes from people who have sent desired email before)

Ratio of email to non-email flows

- Non-email flows are overhead or waste for a spam source
- Don’t care about target information – just trying to pump out spam

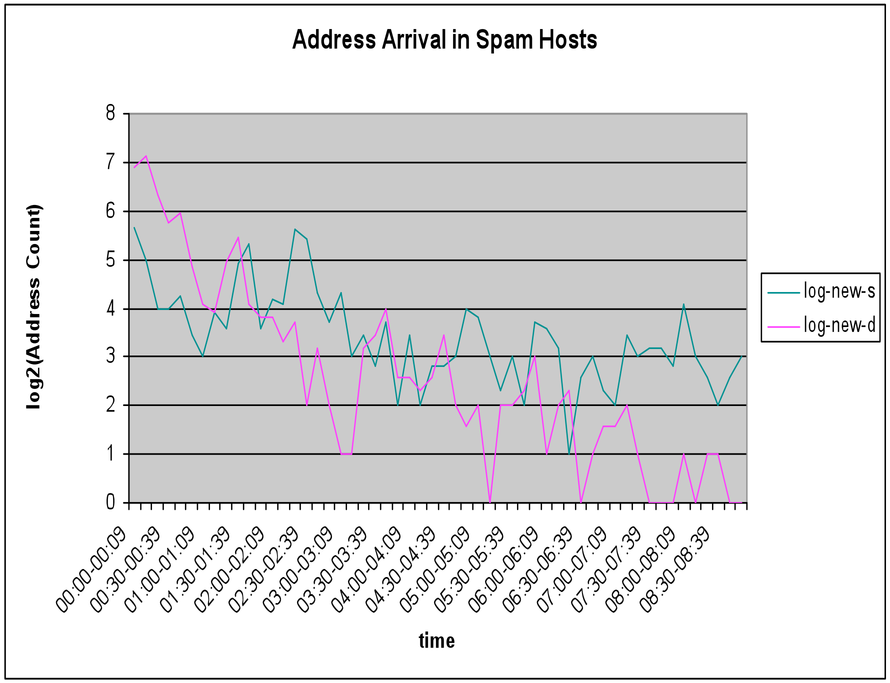
Rate of Sending Email

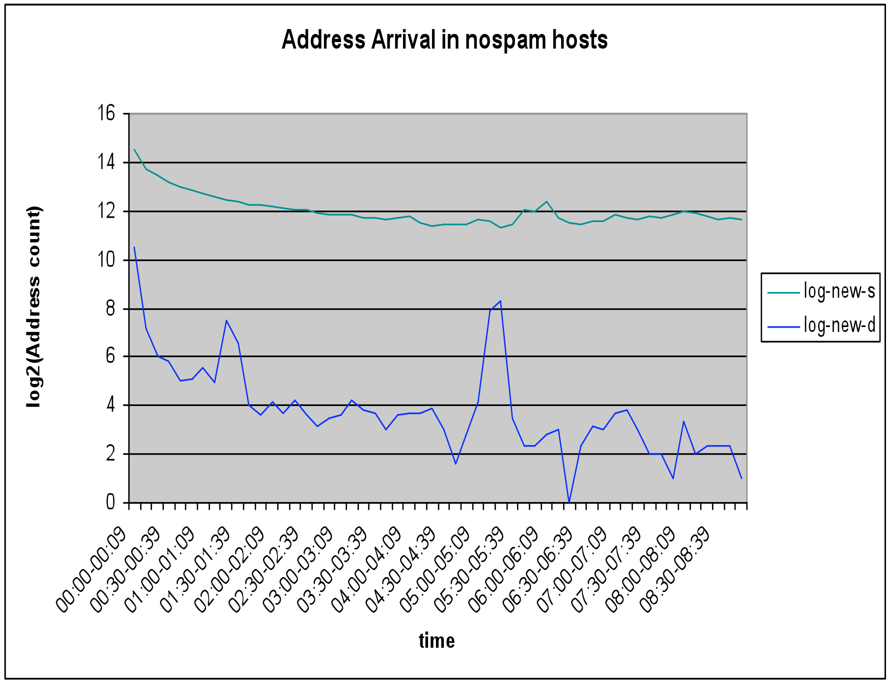




- Spam hosts send at a maximum rate approximately 20 times the maximum rate of non-spam hosts
- Spam hosts send at a mean rate approximately double the maximum rate of non-spam hosts

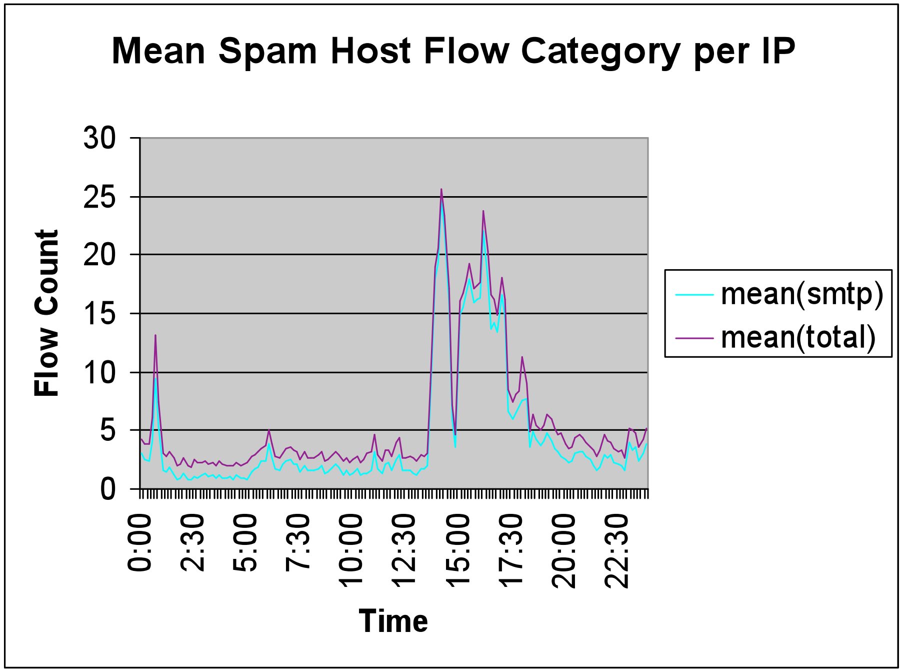
Locality of Addresses Sending Email

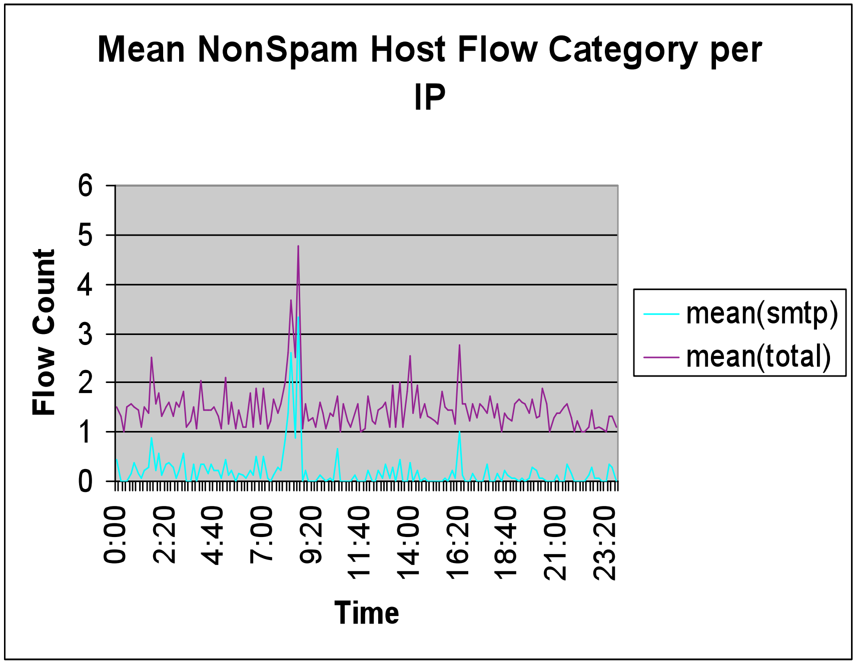




- Much greater address dynamics for spam hosts than for non-spam hosts

Ratio of Mail to Non-Mail





- Much less non-email traffic in spam hosts than in non-spam hosts, particularly during highest peaks

Outline of Spam Source Detector

Locality exploited to construct white list (Frequent or desired email contacts)

Gather list of sources of completed email contacts in 5 minute period, ignoring white list

- Count number of email contacts per source
- Count number of non-email contacts per source

Drop all sources with less than 15 contacts, and all that had at least 10% non-email contacts

- 15 and 10% derived empirically to maximize true positives and minimize false positives

Implementation of Spam Detector

1. Use rwfilter to pull one hour of border-crossing flows, excluding white list
2. Use rwfilter to split into 5-minute flows, then pipe to rwfilter to split into email and non-email flows – each going to a bag
3. Use rwbagtool to limit email bag to at least 15 flows
4. Use rwbagcat on each bag and a python script to drop sources with at least 10% non-email traffic
5. Python script produces list of IP addresses

Runs fairly quickly – but not near-real-time speeds

Validation

Spam sources detected by commercial source

Spam sources detected by email to CERT

Empirical study

- At least as good as commonly-applied spam lists
- Much better than random identification
- Varying the constants has some effect on true positive/false positive rate (although white list helps to limit false positives)

Spam Patterns

Useful for additional confidence

Sources tend to contact in bursts

- Send
- Wait (hours to days)
- Send

Sources tend to be in odd locations

- Not mission-relevant
- DHCP pools
- Multiple sources for similar flow characteristics

Using the spam detection script

```
spamdetector.csh start-date end-date [ip.set]
```

ip.set is for area-of-interest specification

Start-date and end-date are in rwfilter date format

Creates hourly spam source files in current directory

- `spam-yyyy-mm-dd-h.set`
- `spam-yyyy-mm-dd-h.txt`

Responses to Spam

Most spam doesn’t have valid return addresses

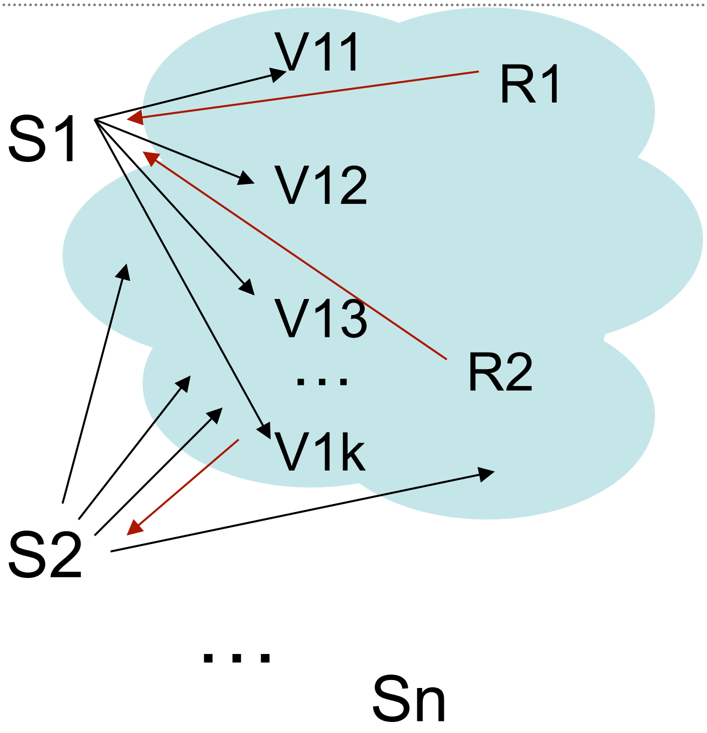
- Click in response
- Stock pump-n-dump scams
- Malicious attachments

… But a small fraction does

- Advance-fee frauds
- Dating frauds
- Some identity theft schemes
- Marketing email

Flow-based spam detection lets us find responses

Process of Detecting Spam Responses



Border data (assume spammers outside organization)

Want to avoid programmed responses

- Response Flows >1hr after spam

Want to only consider complete email connections with content

- Intial: SYN (not SYN-ACK)
- All: SYN-ACK-FIN (not RST)
- 4+ Packets
- 200+ Bytes/Packet average

Want to consider responses only from addresses close to spam destination

- The Tricky Bit!
- A little scripting and PySiLK

The Matching Script

```
import sys
from silk   import *
blocksize=16
try:
        blockfile=file('blocks.csv','r')
except:
        print 'cannot open blocks.csv'
        sys.exit(1)
blockdict=dict()
for line in blockfile:
        fields=line[:-1].strip().split(',')
        if len(fields)<2:
                continue
        try:
                idx = IPAddr(fields[0].strip())
                if idx in blockdict:
                     blockdict[idx].append(IPWildcard(fields[1].strip()+'/'+str(blocksize)))
                else:
                     blockdict[idx]=list([IPWildcard(fields[1].strip()+'/'+str(blocksize))])
        except:
                continue
blockfile.close()
def rwfilter(rec):
       global blockdict
       if (rec.dip in blockdict):
                for pattern in blockdict[rec.dip]:
                        if rec.sip in pattern:
                                return True
        return False
```

Load a dictionary of dip & CIDR/16 pairs to match

Filter based on dictionary

Running the response-detection script

Need to run spam detector first in same directory

```
./findresp.py start-date end-date [ipset] > transcript.txt
```

Start-date and end-date are silk dates

Ipset is for AOR

Produces listing of commands run to generate results

Result files:

- resp-YYYY-M-D.raw
- spam-YYYY-M-D.raw

Can then use silk tools to manipulate flows

Some experience

Large network, 2 days of flows

Spam detected from 643,671 IP addresses

(33M spam flows) requiring approximately 1.5 hours of clock time

Response detected to 1,782 IP addresses (0.28%)

(73,316 response flows) requiring approximately 4 hours of clock time

Example of Spam Responses

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.

Summary

Email is a very valuable communication tool

- Both for us and for the enemy

Flow-based analysis can give us insight on a variety of email-based behaviors, both benign and malicious