Stress Testing Traffic to Infer Its Legitimacy

Nick Duffield
Balachander Krishnamurthy

AT&T Labs-Research, Florham Park, NJ

{duffield, bala}@research.att.com

Impairment: a Fact of Network Life

- Impairment occurs in different protocol/application layers.
- Examples:
  - Transport: packet loss or delay due to congestion
  - SMTP: delayed delivery of email
  - HTTP: request timeout

Recovery from Impairment

- Protocols/Applications/Users routinely recover from impairments
- Examples:
  - Transport: packet loss or delay due to congestion
    - TCP retransmission
  - SMTP: delayed delivery of email
    - Application retry
  - HTTP: request timeout
    - User retrial after some interval (e.g. 1 second)

AT&T

Adaptation to Impairment

- The nature of the adaptation can distinguish good from bad
  - "bad" can mean malicious or anti-social or misconfigured or …
- Examples:
  - Transport: packet loss or delay due to congestion
    - TCP retransmission
    - Well-behaved TCP reduces congestion window as per standards
  - SMTP: delayed delivery of email
    - Application retry
    - User may retransmit mail after notification of delay
    - Spammer less likely to do so
  - HTTP: request timeout
    - User retrial after some human-like interval (e.g. 1 second)
    - DoS attacker prefers to send requests more frequently

Stress Testing: Key Ideas

- Assumptions
  - Differentiation:
    - "Good" and "bad" network traffic responds differently to impairments
  - Recovery
    - Good traffic can tolerate some degree of background impairment
  - Leeway
    - Room to stress by impairment up to level set by SLA
- Proposal
  - Stress test traffic flows with artificial impairments
  - Observe flow's response: helps to classify as good/suspicious/bad
  - Tune level of artificial impairments by cost-benefit analysis
  - Proactive: potentially apply routinely to all traffic

AT&T

Interpretation of Stress Tests

- Combine results with other classifiers
  - Not proposed as a standalone diagnostic
  - e.g. use stress test to move between existing white/gray/blacklist
- Share test results across network
  - Target other stress testers towards suspicious senders
- Robustify classification with multiple tests
  - Fixed horizon: flag as bad if suspicious at least m out on n times
  - Queue-based: flag as bad if suspiciousness is bursty
  - Sequential hypothesis testing
  - Etc..
- Can adapt stress intensity to increase with suspicion level

Stress Testing: Examples

- Transport:
  - Stress: Drop or delay some packets in target flow
  - Test: Observe whether flow response conforms to TCP standard
    - If not, then flag as suspicious
- SMTP:
  - Stress: Delay delivery of email from target mail relay
  - Test: Observe whether email is resent
    - If so, then flag as less suspicious (e.g. move from graylist to whitelist)
- HTTP:
  - Stress: respond with 408 Request Timeout, or 503 Service Unavailable
  - Test: Observe if request repeated at typical human timescales
    - If not, then flag as suspicious

Scales for Acceptable Stress

- Ambient stress level
  - Applications are robust to existing background impairments
  - Design artificial stress characteristics to resemble ambient stress
  - Need good characterization of ambient stress
    - From application level statistics, e.g. server logs
    - From network level statistics, e.g., granular loss, delay statistics
- Service level agreements
  - SLA = limit on total stress
  - Caveat: customers may be acclimated to better "effective" SLA
- Default limit for total stress
  - Stress acceptable if: Artificial Stress + Ambient Stress < SLA

How Much Stress Can the Traffic Take?

- Costs of impairment should not be prohibitive for good traffic
- In some cases, cost of *any* impairment may be too high: avoid
  - highly loss and delay sensitive applications e.g. online gaming
    - Identify (e.g. by application ports) and avoid
  - TCP handshake
    - Identify (by TCP flag) and avoid
- Stress characteristics
  - Frequency, Duration, Granularity
- May want to increase stress in certain circumstances
  - During overflow
  - During attacks

Balancing Total Costs of Stress and Impairment

- Impairment costs
  - Cost to user of impairment
  - Cost to service provider if SLAs violated
- Identification Costs
  - Costs of actions taken on basis of good/suspicious/bad classification
  - False positives (good misclassified as bad)
  - False negative (bad misclassified as good)
- Tune both stress level and actions to minimize total cost
- Tuning of stress can vary spatially and temporally:
  - Adaptive to target response
    - E.g. whitelist good traffic and remove/reduce its stress
  - Adapt to perceived threat level
    - E.g. increase frequency and scope of stress if attack rate increases
  - Stress can also be used as a control action
    - Turn up stress on bad traffic

Scope for Countermeasures

- Well-designed stress test difficult to detect
  - Stress conforms to ambient characteristics
  - Stress tester must use full spectrum of likely impairments
    - E.g. loss and delay in TCP case
    - Suitably randomized to leave no signature
- Method is potentially ubiquitous
  - Makes reverse blacklisting harder
- Aggressive response to impairment not good attack strategy
  - Make flagging as suspicious or bad more likely
- High cost for attacker to try to evade
  - Vs. low impairment cost of stress testing by defender

Relation to Existing Approaches (1)

- Stress Testing originally proposed in TCP by Floyd/Fall (1999)
  - Aim: identify misbehaving flows, penalize to restore fairness
  - Context: unintentional misbehavior due to bad implementation
- Our focus is on deliberate attacks
  - Surviving attacks takes precedence over fairness
    - as opposed to fairness for all
  - Advocate applying routine to any flow
    - Rather than waiting for an attack
- Proposed methods for inference of TCP response
  - Inference of TCP congestion window by Jaiswal et. al. (2004)
  - Uses passive measurements in middle network
  - Accommodates TCP variants
  - Potential to exploit for stress testing
    - Somewhat easier: measure at target, eliminate some uncertainty

Relation to Existing Approaches (2)

- Honeypots
  - Operating at various levels ranging from kernel to application
  - Operating in unadvertised address spaces:
    - any sender in this space is flagged as bad
- Email:
  - Puzzles used to distinguish human senders
- P2P
  - Impairment (tit-for-tat tailoring of upload bandwidth) popular in eMule/BitTorrent P2P networks to prevent freeloaders

Further Work

- So far: framework with potential applications
- First planned evaluation:
  - TCP case
  - Controlled TCP senders configured to act on good or bad manner
  - Stress testing by loss/delay of packets at receiver
  - Classification based on inferred congestion window

Stress Testing: Summary

- Stress testing of traffic
  - Stress test traffic with artificial impairments
  - Help classify as good/bad based on response
  - Stress level comparable with ambient stress and SLAs
    - Stress within expected limits to which good traffic can adapt
  - Tune/adapt stress level, according to
    - Costs of misclassification
    - Perceived threat level
    - Historical response of traffic entity to stress testing
- Potential ubiquitous use
  - Applicable at different application/protocol levels
    - E.g. TCP, SMTP, HTTP, P2P
  - Low cost routine application
  - Difficult to detect and counter