Deep Packet Inspection (DPI)

- Effective intrusion detection requires accurate traffic inspection
- In practice: Deep Packet Inspection (DPI)
Deep Packet Inspection (DPI)

- Effective intrusion detection requires accurate traffic inspection

- In practice: Deep Packet Inspection (DPI)
  - cannot rely on port/protocol assumptions alone
Deep Packet Inspection (DPI)

- *Effective* intrusion detection requires *accurate* traffic inspection

- In practice: *Deep Packet Inspection (DPI)*

  - cannot rely on port/protocol assumptions alone

  - must scale to cope with:
    - massive traffic volumes
    - significant heterogeneity of traffic
Deep Packet Inspection (DPI)

- Effective intrusion detection requires accurate traffic inspection

- In practice: Deep Packet Inspection (DPI)
  - cannot rely on port/protocol assumptions alone
  - must scale to cope with:
    - massive traffic volumes
    - significant heterogeneity of traffic
  - leads to common trade-off: accuracy vs. resources
opaque: compressed or encrypted
opaque: compressed or encrypted

- DPI systems cannot directly derive useful information from opaque data packets:
DPI systems cannot directly derive useful information from opaque data packets:

- compressed: require decompression
opaque: compressed or encrypted

- DPI systems cannot directly derive useful information from opaque data packets:
  - compressed: require decompression
Opaque Traffic

opaque: compressed or encrypted

- DPI systems cannot directly derive useful information from opaque data packets:
  - compressed: require decompression
  - encrypted: ???
- **Opaque payload bytes are effectively random**
- signature matches unlikely
- **slow path**: every packet compared against every signature
DPI Engines and Opaque Traffic

- Opaque payload bytes are effectively random
- signature matches unlikely
- slow path: every packet compared against every signature
- Encrypted packets: CPU overhead several orders of magnitude higher (Cascarano et al, 2009)
Prevalence

- Encrypted:
  - HTTPS
  - VPN connections
- consider: private corporate networks
Prevalence

- **Encrypted:**
  - HTTPS
  - VPN connections
  - consider: *private corporate networks*

- **Compressed:**
  - streaming audio/video
  - most images (JPEG, PNG)
  - many HTML websites
Empirical Observations

- Surprising *preponderance of opaque traffic*
- (payload-carrying) TCP packets: *89%*
- corresponding to *86%* of payload bytes
Empirical Observations

- Surprising *preponderance of opaque traffic*
- (payload-carrying) TCP packets: 89%
- corresponding to 86% of payload bytes

- As a community, we *must* adapt to cope with opaque traffic
- *idea*: partition traffic into classes for specialized processing
- *first steps*: fast, accurate *winnowing* (i.e., filtering) of opaque traffic
Our Contributions

- Identification of *opaque traffic* as an important and distinguishable class of network traffic
Our Contributions

- Identification of **opaque traffic** as an important and distinguishable class of network traffic
- Development, comparison, and evaluation of **multiple techniques for quickly and accurately identifying opaque packets**
Our Contributions

* Identification of *opaque traffic* as an important and distinguishable class of network traffic

* Development, comparison, and evaluation of *multiple techniques for quickly and accurately identifying opaque packets*

* An *operational analysis* of modern network traffic with respect to opacity
Our Contributions

- Identification of *opaque traffic* as an important and distinguishable class of network traffic
- Development, comparison, and evaluation of *multiple techniques for quickly and accurately identifying opaque packets*
- An *operational analysis* of modern network traffic with respect to opacity
- Evaluation, at scale, of the potential for *winnowing* to reduce load on IDS/DPI systems
Identifying Opaque Traffic

Signatures?

secure sockets layer
transport layer security
secure shell
message stream encryption
Identifying Opaque Traffic

Signatures?

* requires construction and deployment of signatures for each and every protocol
Identifying Opaque Traffic

Signatures?

- requires construction and deployment of signatures for each and every protocol
- some opaque protocols designed to evade signatures (e.g., BitTorrent’s Message Stream Encryption)
Identifying Opaque Traffic

Content-Type inspection?
Content-Type inspection?

- requires *flow reassembly*
- 1/3 of runtime overhead in our experiments
Identifying Opaque Traffic

Content-Type inspection?

- requires *flow reassembly*
- 1/3 of runtime overhead in our experiments
- often *inaccurate*
- demonstrated later
- independently corroborated *(Schneider et al, 2012)*
Our Design Criteria

* per-packet operation
Our Design Criteria

- *per-packet* operation
- *no reassembly required!*

Andrew M. White  Clear and Present Data  10
Our Design Criteria

- *per-packet* operation
- *no reassembly required!*
- flows can change opacity:
  - *gzipped HTTP*
  - *STARTTLS*
Our Design Criteria

- *per-packet* operation
- *no reassembly required!*
- flows can change opacity:
  - *gzipped HTTP*
  - *STARTTLS*
- *port- and protocol-agnostic*
Our Design Criteria

- *per-packet* operation
- *no reassembly required!*
- flows can change opacity:
  - *gzipped HTTP*
  - *STARTTLS*
- *port- and protocol-agnostic*
- *minimal payload inspection*
- resource use increases with inspection depth (*Dreger et al, 2004, Cascarano et al, 2009*)
Our Techniques

❖ Small-sample hypothesis tests
❖ extensive experimentation (details in the paper)
❖ comparison of methods
❖ parameter-space exploration
Our Techniques

- Small-sample hypothesis tests
- extensive experimentation (details in the paper)
  - comparison of methods
  - parameter-space exploration
- Two clearly superior methods
  - Likelihood Ratio
  - (Truncated) Sequential Probability Ratio Test (SPRT)
Our Techniques

❖ Small-sample hypothesis tests
❖ extensive experimentation (*details in the paper*)
❖ comparison of methods
❖ parameter-space exploration

❖ Two clearly superior methods
❖ *Likelihood Ratio*
❖ *(Truncated) Sequential Probability Ratio Test (SPRT)*

❖ Identify opaque packets in *16 bytes or less*
❖ significantly fewer than necessary for, e.g., entropy, chi-square
Major Experiments

- File Type Opacity
- Content-Type Matching
- Operator Analysis
- Head-to-Head Comparison
Content-Type Matching

- Logged traffic using **Bro**
- ports 22, 25, 80, 443
- two major university campuses
- *dynamic protocol detection*
Content-Type Matching

● Logged traffic using *Bro*
  ● ports 22, 25, 80, 443
  ● two major university campuses
  ● *dynamic protocol detection*

● *ground truth:*
  ● SSL/TLS and SSH: *opaque*
  ● SMTP: *transparent*
  ● HTTP: inferred from Content-Type and Content-Encoding
Match Rate

University of Michigan (39m packets; 3.8m flows)

University of North Carolina (24m packets, 2.3m flows)
Figure 7: ROC plots for the techniques examined in this work as accurately as those in the byte value domain; we believe that this indicates that accurate entropy tests require more samples than are available in our context.

Content Type Matching

We performed a large-scale analysis on both trace1 and trace2; the overall results are given in Table IV. Using the truncated sequential method \( v = 2.27 \), \( \tau = 2.27 \), \( \tau = 2.07 \), \( N = 38 \), \( T = 0 \) we achieve a match rate ifle the percentage of examples for which our techniques produced the same label as expected from the content type of .7s on trace1 and .8s on trace2. We refer to 'match' rates here rather than false positive or false negative rates due to the large quantity of mislabeled content types we encountered. In the case of encrypted traffic (like SSL/TLS) we accurately classified approximately .7s of the traffic. However, the mismatches are particularly interesting. Figure 8 shows the distribution of packet IDs where a packet's ID is its position in the bidirectional flow (a packet with ID zero is the first packet sent by.

Figure 8: CDFs of Packet IDs

- mismatches: flagged as transparent (“false negatives”)
- 95% of mismatches within first 5 packets of flow
- primarily connection-setup packets
HTTP Text Mismatches

text/plain, labeled \textit{transparent}

text/plain, labeled \textit{opaque} ("false positives")
HTTP JPEG Mismatches

image/jpeg, labeled *opaque*

image/jpeg, labeled *transparent* ("false negative")
HTTP JPEG Mismatches

image/jpeg, labeled *opaque*

image/jpeg, labeled *transparent* (“false negative”)
Head to Head

- Implemented *winnowing* as a *Snort* preprocessor
Head to Head

- Implemented **winnowing** as a **Snort** preprocessor
- Ran two **Snort** instances side-by-side on live traffic
  - one had our preprocessor installed
  - both saw *exactly the same packets*
Head-to-Head Experiment

- DAG (Data Acquisition and Generation) capture card
- packet duplication
- port filtering
Head-to-Head Experiment

- DAG (Data Acquisition and Generation) capture card
- packet duplication
- port filtering
- One experiment:
  - 24 weekday hours
  - peak load of 1.2Gbps
  - nearly 100 billion packets
  - 7.6 terabytes of data
Packets Analyzed

![Graph showing analyzed packets over time]

- **Y-axis**: Analyzed Packets
- **X-axis**: Time
- **Legend**:
  - `stock snort`
  - `winnnowing`
Network Exposure Difference

- Attempted User Privilege Gain
- Misc activity
- Potentially Bad Traffic
- A Client was Using an Unusual Port
- A Network Trojan was Detected
- Attempted Administrator Privilege Gain
- Potential Corporate Privacy Violation
- Unknown Traffic

Percent Difference (Winnow vs. Snort) 0% 500% 1000% 1500% 2000%
Network Exposure Difference

- Attempted User Privilege Gain
- Misc activity
- Potentially Bad Traffic
- A Client was Using an Unusual Port
- A Network Trojan was Detected
- Attempted Administrator Privilege Gain
- Potential Corporate Privacy Violation
- Unknown Traffic

Percent Difference (Winnow vs. Snort) 0% 500% 1000% 1500% 2000%
Network Exposure Difference

- Attempted User Privilege Gain
- Misc activity
- Potentially Bad Traffic
- A Client was Using an Unusual Port
- A Network Trojan was Detected
- Attempted Administrator Privilege Gain
- Potential Corporate Privacy Violation
- Unknown Traffic

Percent Difference (Winnow vs. Snort) 0% 500% 1000% 1500% 2000%
Summary

- **Opaque traffic**: compressed or encrypted network traffic
- Surprisingly high proportion (89% packets; 86% of payload)
- Evaluated multiple techniques for identifying opaque packets
Summary

❖ **Opaque traffic**: compressed or encrypted network traffic

❖ surprisingly high proportion (89% packets; 86% of payload)

❖ evaluated *multiple techniques* for identifying opaque packets

❖ Explored *winnowing* (i.e., filtering) opaque packets

❖ first step toward coping with opaque traffic
  • improves *accuracy vs. resources* curve (more signatures can be applied to transparent traffic)

❖ *not* a solution by itself, but a tool in the toolbox
