

# Effective Detection of Multimedia Protocol Tunneling using Machine Learning

Diogo Barradas, Nuno Santos, Luís Rodrigues  
INESC-ID, Instituto Superior Técnico, Universidade de Lisboa

USENIX Security Symposium '18



# Internet Censorship is Widespread Around the Globe

## Russia begins blocking access to Telegram

TUE, APR 17, 2018 - 7:00 AM

### After John Oliver's Jokes About Xi Jinping, China Blocks HBO Website

A week after Mr. Oliver mocked China's president, the authorities there have taken additional strides to block him on the internet.

By TIFFANY MAY and OLIVIA MITCHELL RYAN

## Turkey Gives Its Aggressive TV Control Over the Web

By Firat Kozok

22 de março de 2018 10:42 GMT

## No Justification for Spanish Internet Censorship During Catalanian Referendum

BY JEREMY MALCOLM | OCTOBER 2, 2017

13:34 27 Dec 2017

### Vietnam censors to fight 'internet chaos'



Local media reports the People's Army has hired more than 10,000 people to tackle "wrongful views".

[Read more >](#)



12:00 21 Feb

[Rajab activist jailed over tweets](#)

ASIA 14 June 2018

### The perils of being a journalist in Modi's India

Cases of Rana Ayyub, Barkha Dutt and Ravish Kumar, who are critical of the government, highlight lack of press freedom.

! Rajab has been sentenced to five years in prison for ing the government on Twitter.

[I more >](#)

May 1, 2018

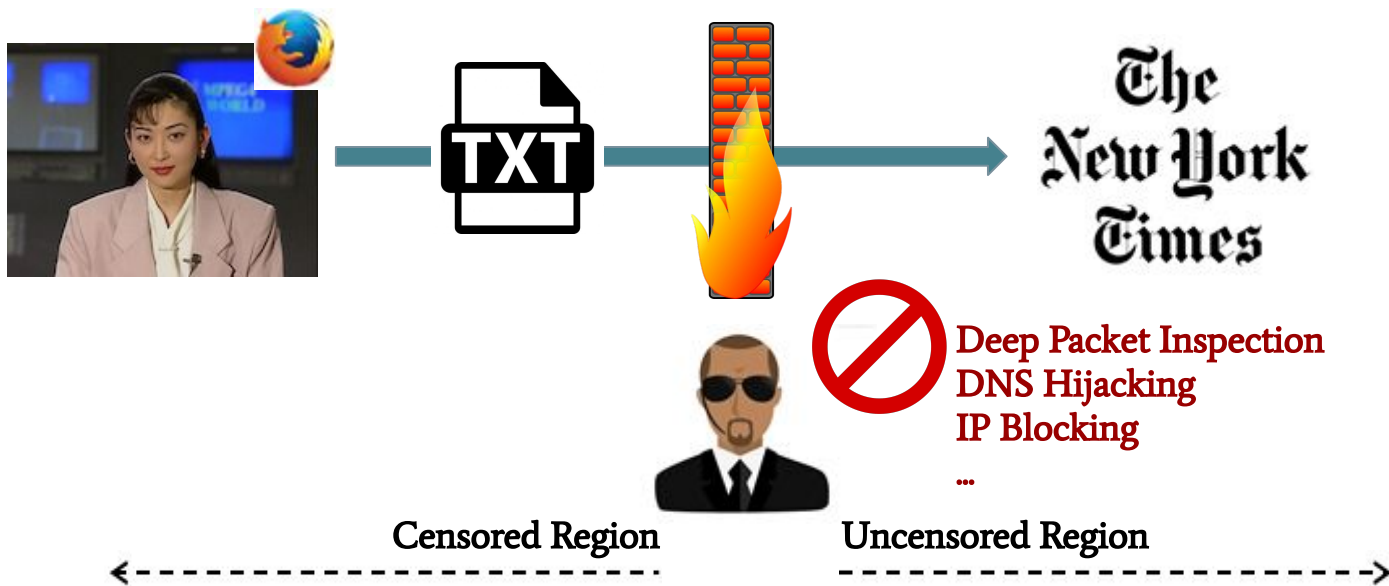
### Iran, Like Russia Before It, Tries to Block Telegram App

The encrypted social media app is the service restrictive governments love to hate, but they are often powerless to shut it down entirely.

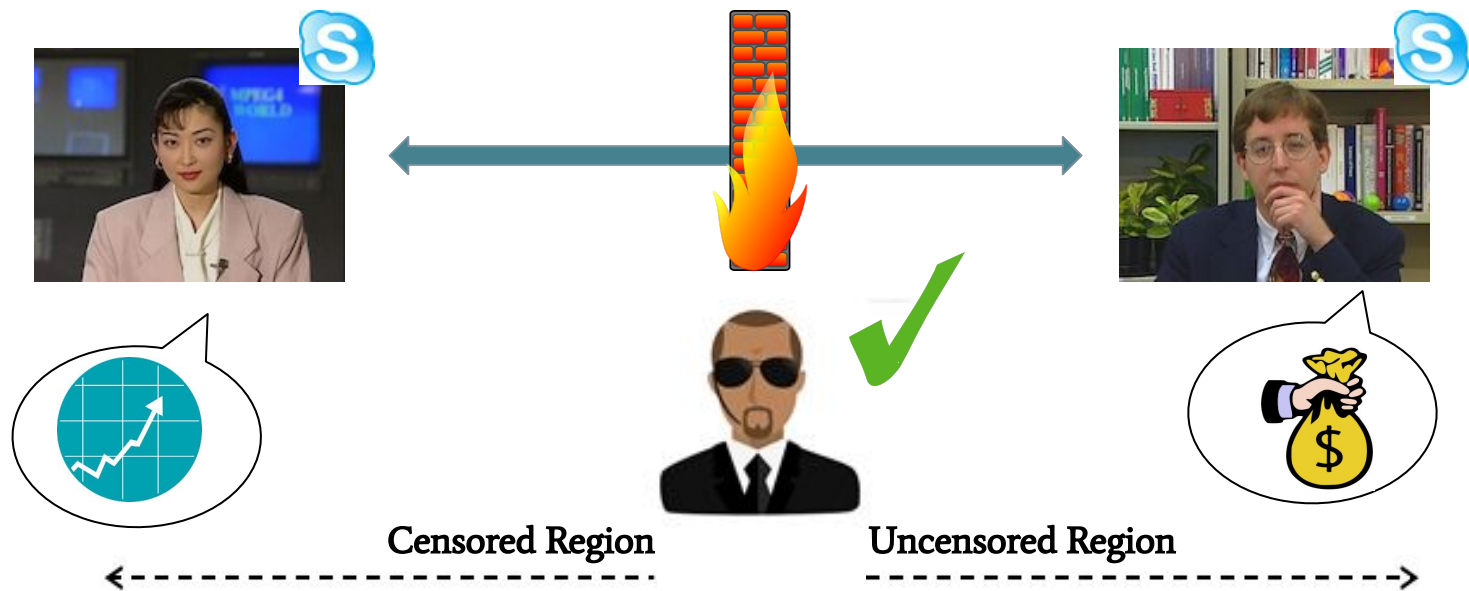
By THOMAS ERDBRINK



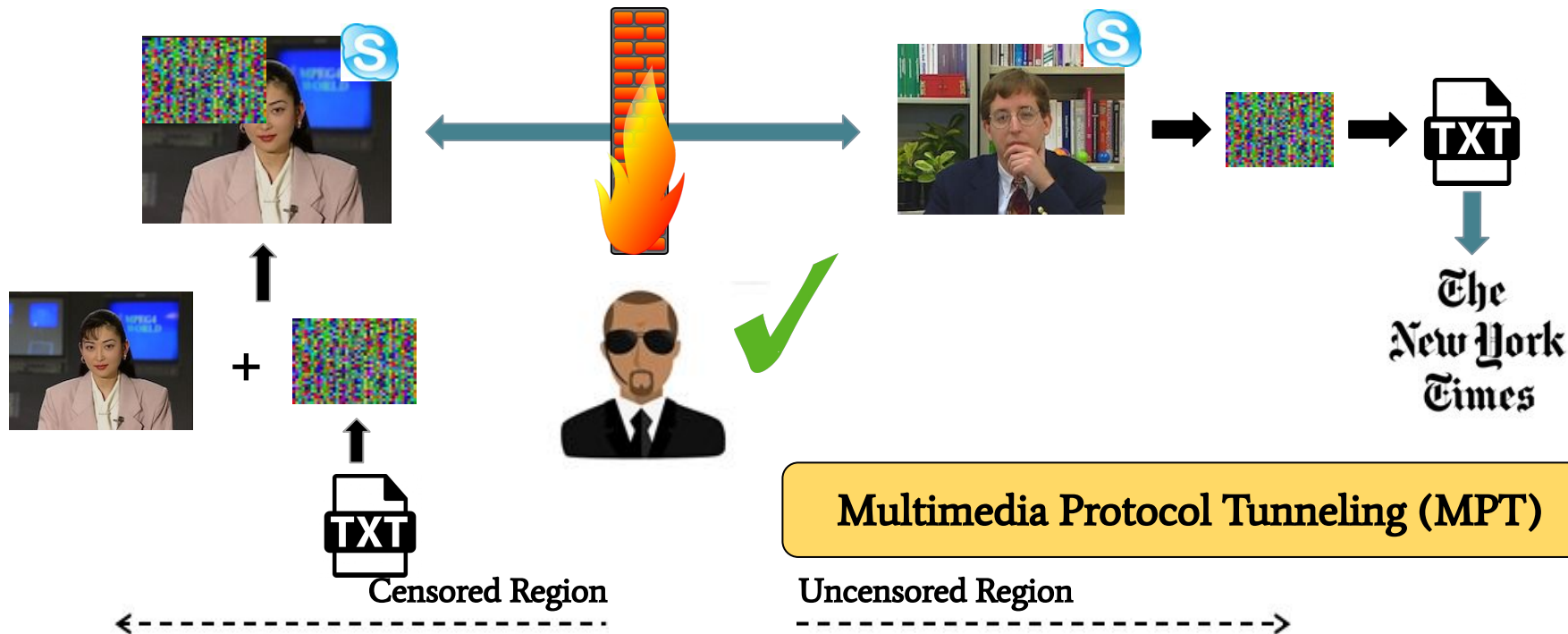
# Adversaries Monitor and Control Internet Access



# But Some Communication Channels are Still Allowed...

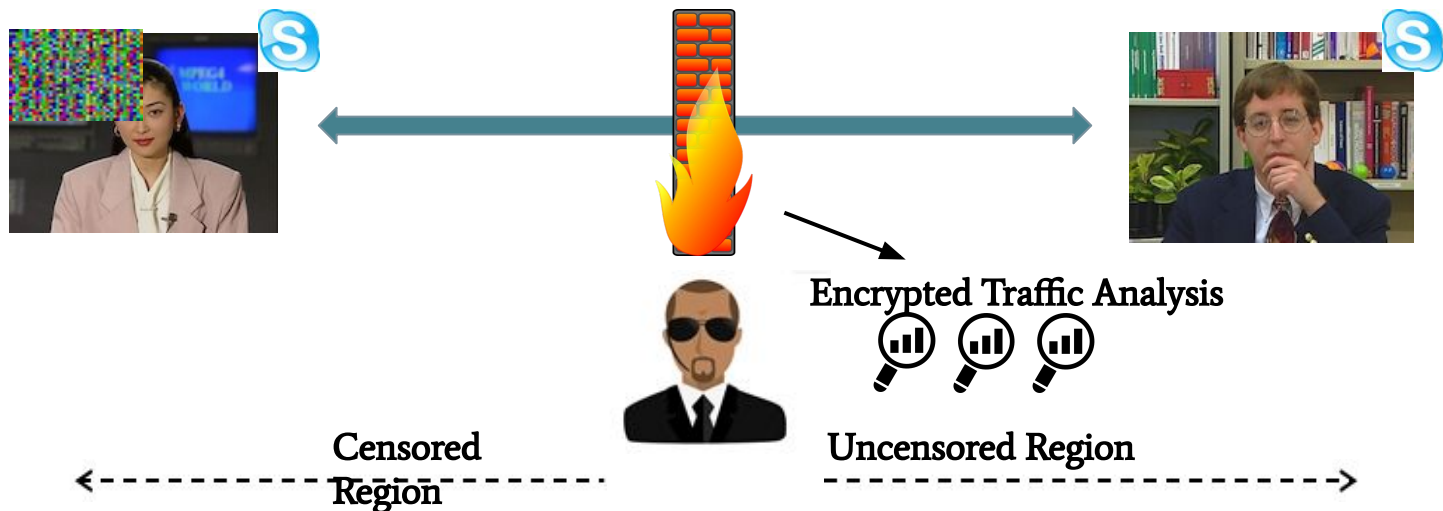


# How Can We Encode Data into Multimedia Streams?



# Is MPT a Silver Bullet for Evading Censorship?

- **Traffic characteristics change with covert data embedding**
  - Due to changes in video compression
- **Adversaries can detect unusual patterns in encrypted network flows**
  - Comparison of packets' size / inter-arrival time probability distributions



# Unobservability

A covert channel is **unobservable** if an adversary cannot distinguish streams that carry a covert channel from those that do not

- Previous works have attempted to assess unobservability using **simple similarity-based classifiers**
  - Results suggested that covert channels were unobservable
- **Are these claims sound?**
  - If not, they pose life-threatening risks to end users (e.g. journalists)

# Limitations of Previous Work

- Different works use **different evaluation metrics**
- Unobservability assessment is based on a **limited set of features**
- **Do not** leverage recent advances on ML techniques



# Goal

**To understand whether state-of-the-art MPT systems are  
secure**

**Provide an answer to the following questions:**

- How effective existing detection techniques really are?
- Are there classification techniques which offer better results?
- Which features can be used to better detect covert channels?

# Contributions

The first extensive experimental study of the unobservability of covert channels produced by state-of-the-art MPT systems

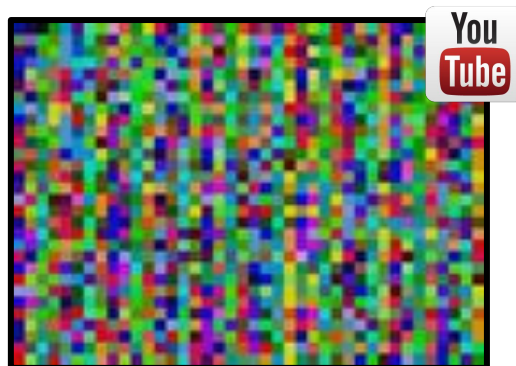
- **Compare existing similarity classifiers on the detection of MPT tools**
  - In general, unable to accurately detect covert channels
- **Explore multiple ML techniques for the detection of covert channels**
  - Decision tree-based classifiers can effectively detect existing MPT tools
- **Analyse the importance of multiple features for MPT detection purposes**
  - We find that packet lengths matter the most

# Multimedia Protocol Tunneling (MPT) Systems Under Study



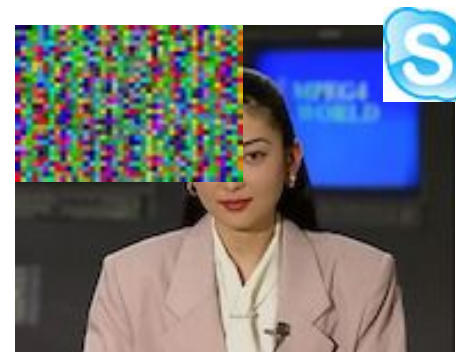
**Facet (WPES'14)**

Unidirectional (A/V)  
Video Transmission



**CovertCast (PETS'16)**

Unidirectional (V)  
Censored Websites Transmission



**DeltaShaper (PETS'17)**

Bidirectional (V)  
Arbitrary Data Transmission

# Multimedia Protocol Tunneling (MPT) Systems Under Study



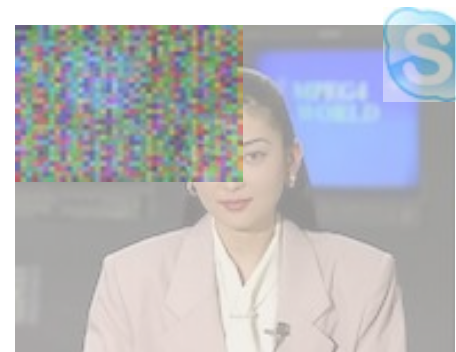
**Facet (WPES'14)**

Unidirectional (A/V)  
Video Transmission



**CovertCast (PETS'16)**

Unidirectional (V)  
Censored Websites Transmission



**DeltaShaper (PETS'17)**

Bidirectional (V)  
Arbitrary Data Transmission

# Adversary Model

- **We emulate a passive state-level adversary**
  - Able to inspect encrypted traffic streams
  - Uses multiple anomaly detection techniques
- **Adversary cannot decrypt the encrypted traffic streams**
  - And cannot control the software at users' endpoints
- **Adversary does not collude with multimedia applications' providers**

**The adversary cannot disclose raw multimedia content**

# Experimental Setup

- **Local Testbed**

- Two Ubuntu VMs on 2.4 GHz Intel Core2 Duo CPU, 8GB RAM
  - Skype for Web, YouTube (QUIC packets)
  - MPT tools prototypes

- **Dataset**

- **Facet:** 1000 chat streams, 1000 covert streams
- **DeltaShaper:** 300 chat streams, 300 covert streams
- **CovertCast:** 200 live streams, 200 covert streams

- **Network packets are collected for 60 seconds**

# How are Covert Channels Detected Today?

- **Previous systems were evaluated with different similarity-based classifiers**
  - **Facet** : Pearson's Chi-squared Test ( $\chi^2$ )
  - **CovertCast** : Kullback-Leibler Divergence (KL)
  - **DeltaShaper** : Earth Mover's Distance (EMD)
- **Feature sets are similar (quantized frequency distributions)**
  - **Facet** : Packet size bi-grams
  - **CovertCast** : Packet size, inter-arrival delay
  - **DeltaShaper** : Packet size, inter-arrival delay

# How Effective are Existing Detection Techniques?

Protocol Tunneling System	$\chi^2$ Classifier (acc%)	KL Classifier (acc%)	EMD Classifier (acc%)
Facet ( $s = 50\%$ )	74.3	57.5	57.5

$\chi^2$  is the most accurate classifier

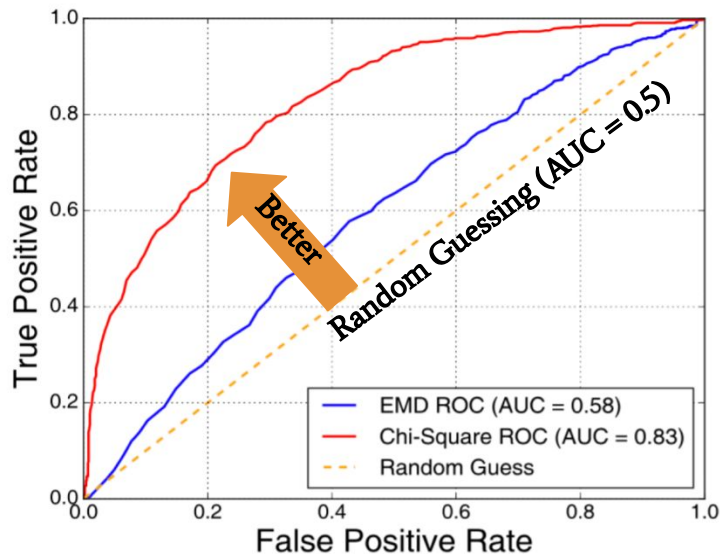
KL and EMD are comparable  
Recent classifiers offer worse

accuracy

- But adversaries are interested in checking TPR and FPR trade-offs
  - Flag as many covert channels as possible
  - Erroneously flag few legitimate connections



# Similarity-based Classifiers Produce a Large FPR



Facet:

$\chi^2$ : 90% TPR = **45% FPR**

EMD: 90% TPR = **84% FPR**

Can we do  
better?

**Similarity-based classifiers are not suitable for the effective detection of**  
**Facet**

# Can Other ML Techniques Better Detect Covert Channels?

- **We assess the effectiveness of multiple decision tree-based classifiers**
  - Decision Trees
  - Random Forests
  - **eXtreme Gradient Boosting (XGBoost)**
- **Models are easily interpretable**
- **Provide the ability to assess feature importance**

kaggle



# Which Features Could an Adversary Use?

- **Feature set 1: Summary statistics (ST)**

- Total of 166 features, including simple statistics (e.g., max, min, percentiles), high order statistics (e.g., skew), and bursts

**A set of features not previously considered for the detection of MPT tools**

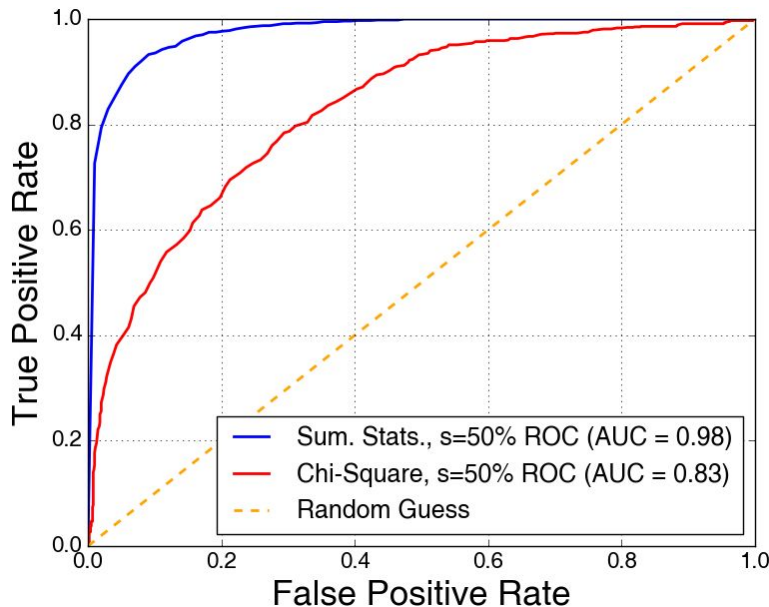
- **Feature set 2: Quantized packet lengths (PL)**

- Quantized PL frequency distribution for the flow carrying covert data
- Each K size bin acts as an individual feature (K = 5 bytes)

**Decision trees exploit the different relevance of particular ranges of this feature**

**set**

# XGBoost + ST is Effective at Detecting Covert Channels



Facet:

XGBoost-ST: 90% TPR = **7% FPR**



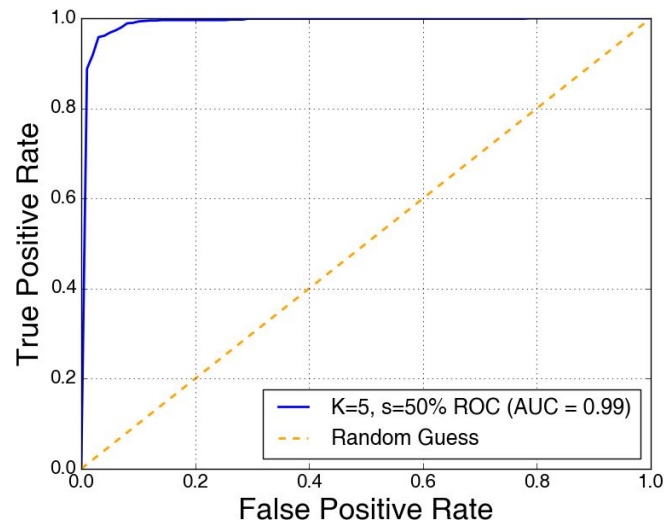
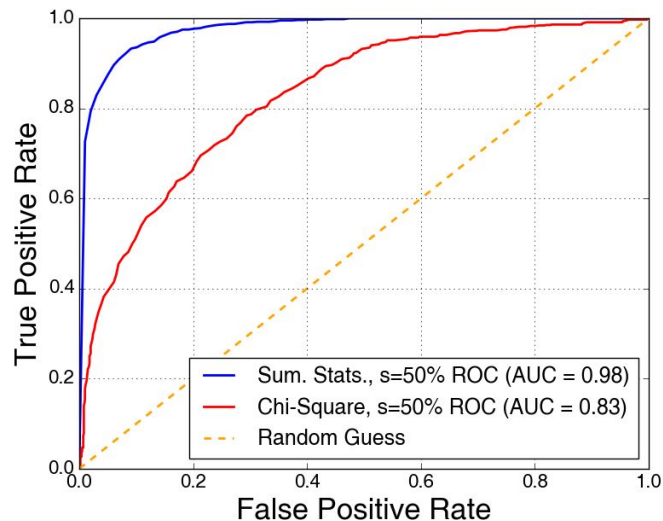
**Majority of covert channels can be flagged  
with a small number of false positives**

$\chi^2$ : 90% TPR = **45% FPR**



**XGBoost-ST offers a much lower FPR than  $\chi^2$**

# Quantized Packet Lengths Outperform Summary Statistics

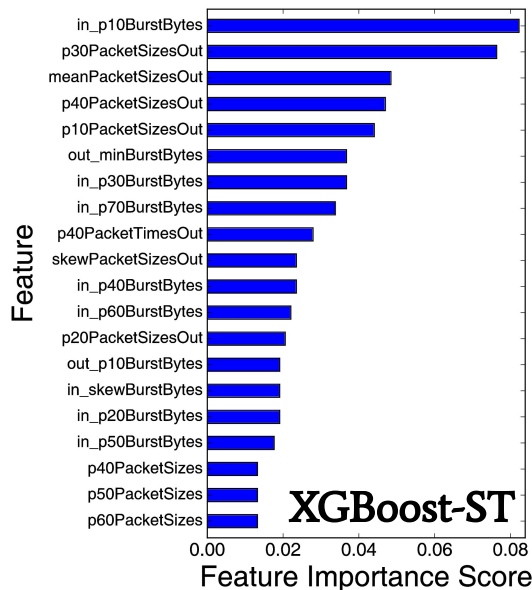


**XGBoost-ST:** 90% TPR = **7% FPR**

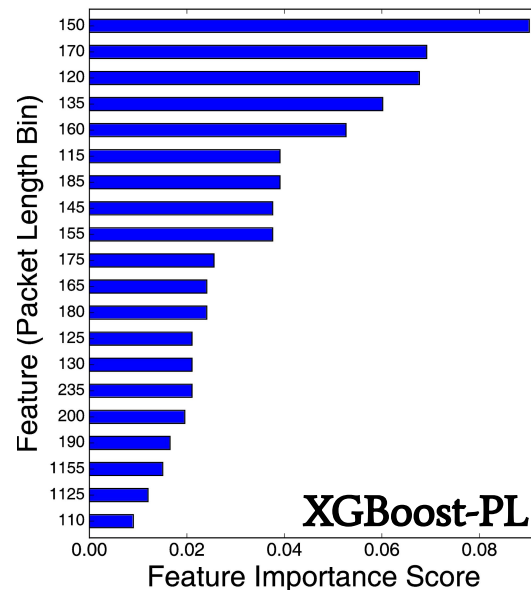
**XGBoost-PL:** 90% TPR = **2% FPR**

**XGBoost-PL reduces the FPR when flagging the same amount of covert channels**

# Which Features can Better Identify Facet Traffic?




**ST features indicate that PL features are more relevant**



**PL within 100-200B is most relevant for detection. This is the typical range for audio data**

# Is Detection Effective Without a Fully Labeled Dataset?

- **XGBoost is effective, but requires a fully labeled dataset**
  - Both legitimate and covert traffic samples
- **Adversaries may be unable to synthesize covert traffic**
  - e.g., difficulty in obtaining a particular MPT tool, “zero-day” MPT tools
- **We investigate the effectiveness of semi- and unsupervised ML techniques**
  - OCSVM
  - Autoencoder
  - Isolation Forest

**Train with legitimate samples only**

**Train without labeled samples**

# Labeled Data is Required for Accurate Detection

Protocol Tunneling System	XGBoost-PL (AUC)	Autoencoder (AUC)	OCSVM (AUC)	Isolation Forest (AUC)
Facet ( $s = 50\%$ )	0.99	0.70	0.63	0.56



Autoencoders are promising for the identification of covert traffic



OCSVMs have limited capability for covert traffic detection



Isolation Forest is unable to accurately detect covert channels

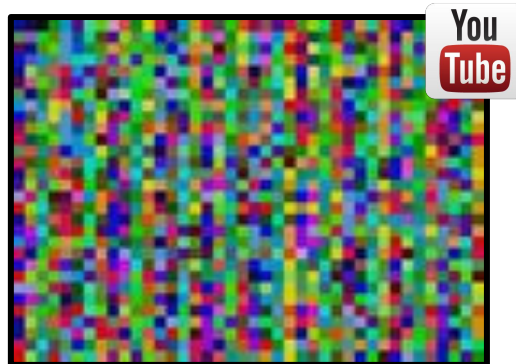


# Overview on the Detection of Other MPT Tools



**Facet (WPES'14)**

Unidirectional (A/V)  
Video Transmission



**CovertCast (PETS'16)**

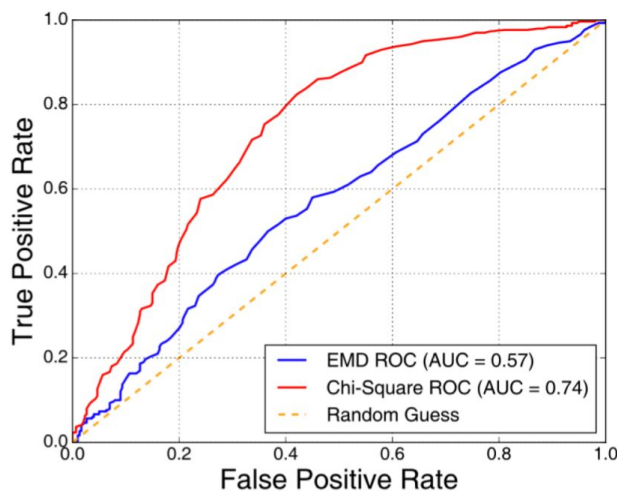
Unidirectional (V)  
Censored Websites Transmission



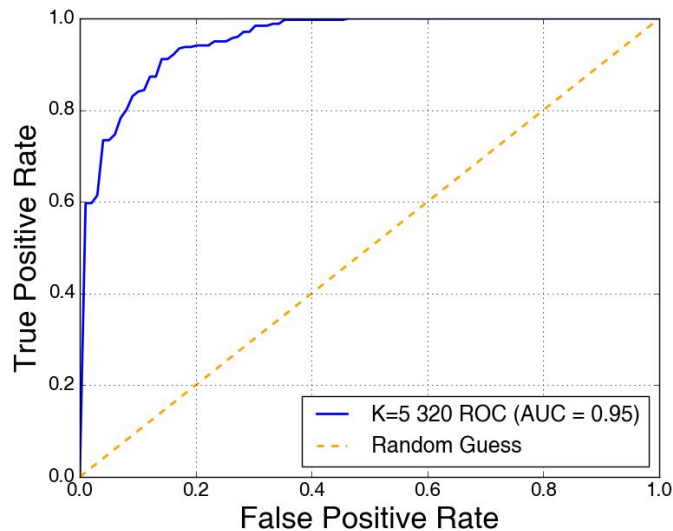
**DeltaShaper (PETS'17)**

Bidirectional (V)  
Arbitrary Data Transmission

# Detection of DeltaShaper



$\chi^2$ : 90% TPR = **51% FPR**



**XGBoost-PL**: 90% TPR = **14% FPR**

**DeltaShaper detection results follow a similar trend to those of Facet detection**

# Detection of CovertCast

Protocol Tunneling System	$\chi^2$ Classifier (acc%)	KL Classifier (acc%)	EMD Classifier (acc%)
CovertCast	99	92	83

**CovertCast can be easily detected by similarity-based classifiers**

- **How tightly are MPT designs coupled to carrier protocols / applications?**
  - Results suggest that implementation changes on carrier protocols can affect the unobservability of covert channels

# Conclusions

- Existing MPT tools can be effectively detected despite previous unobservability claims provided by similarity-based classifiers
- The analysis of feature importance offers potentially actionable data for the development of new MPT designs
- More sophisticated semi-supervised ML techniques are promising for the detection of MPT tools

<https://web.ist.utl.pt/diogo.barradas>

