

Analyzing an Adaptive Reputation Metric for Anonymity Systems

**Anupam Das (UIUC),
Nikita Borisov (UIUC),
Matthew Caesar (UIUC)**



**Hot SoS
2014**



Anonymous Communication

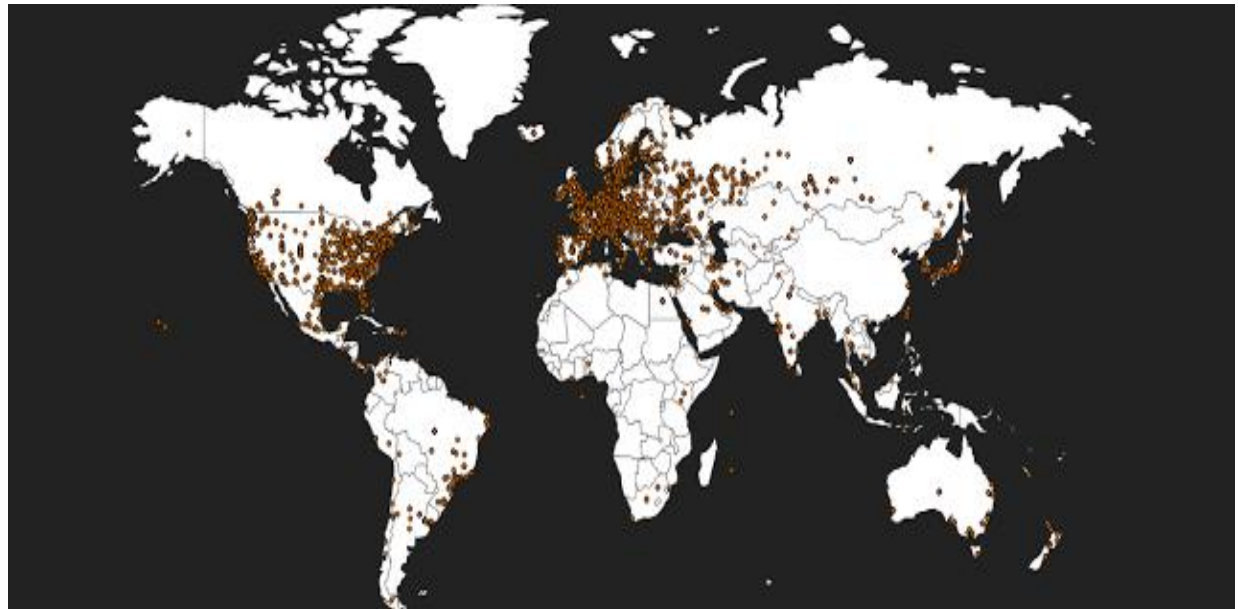
Hides user identity and defends users against **internet surveillance** and **traffic analysis**. **Tor** is a system that provides online anonymity. It is one of the most widely used anonymity network.



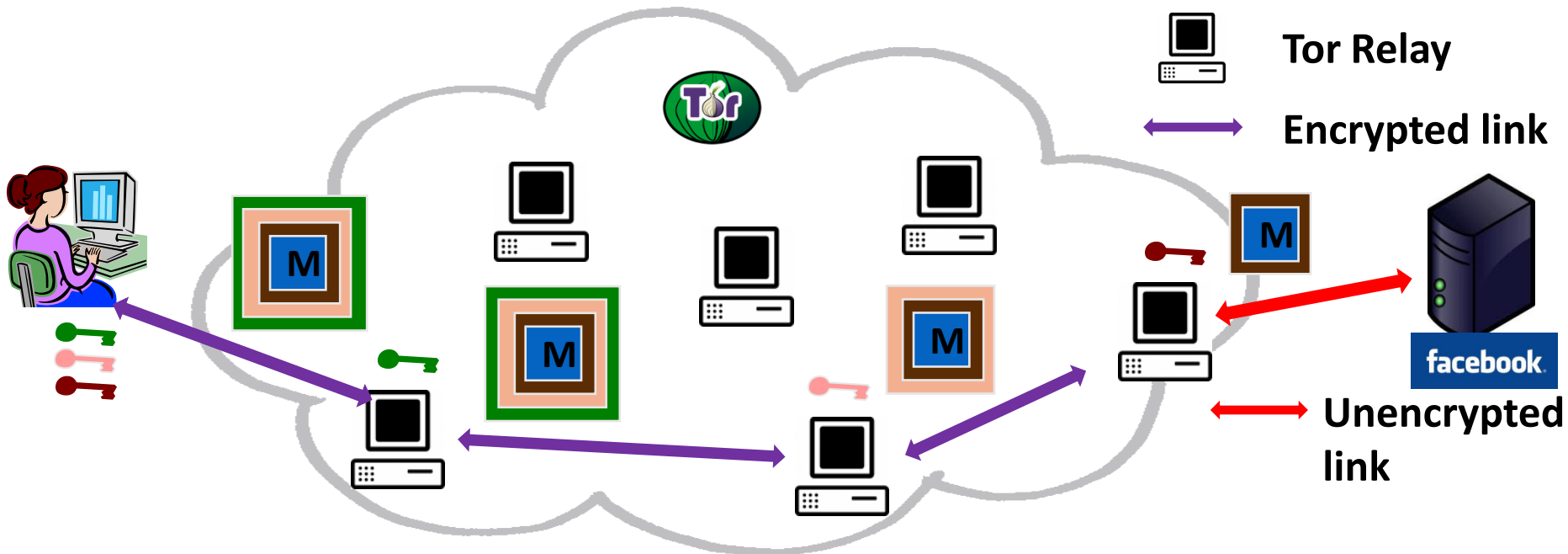
~5000 Tor Relays

~300,000 Users daily

<https://metrics.torproject.org>



How Tor Works?

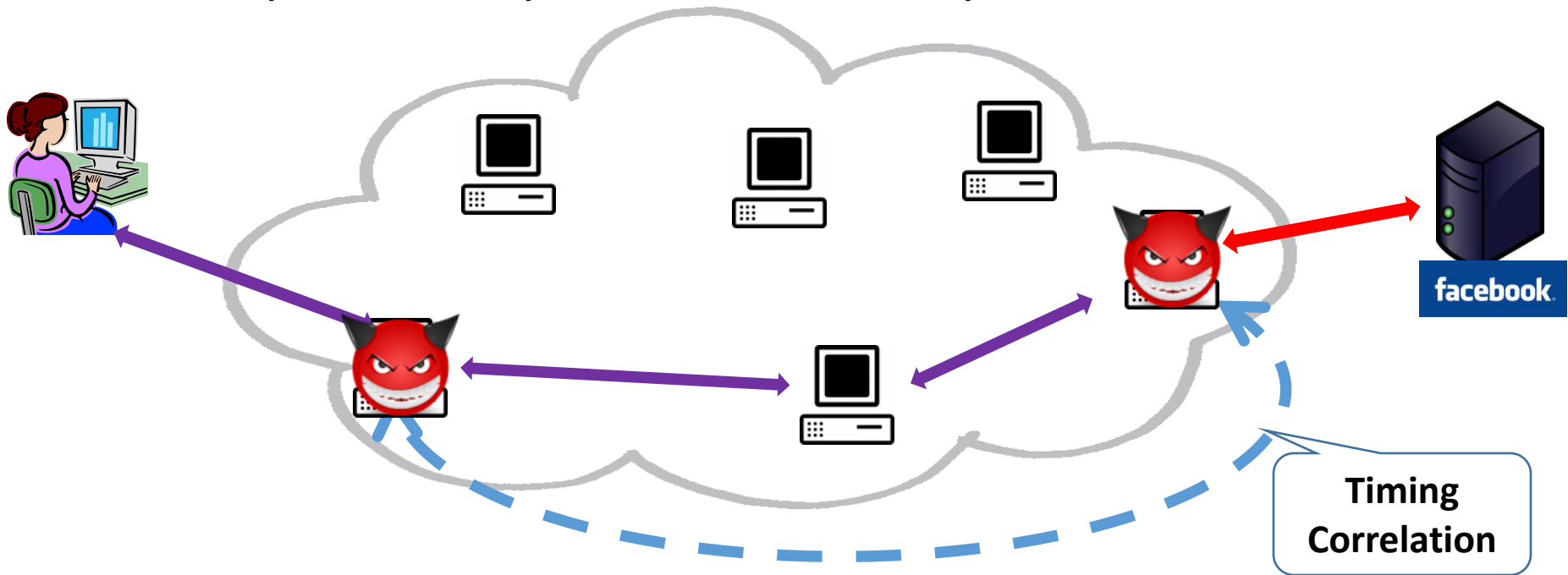


- Tor circuit /tunnel is built incrementally one hop by one hop
- Layered encryption is used
 - Each router knows only its predecessor and successor



Threats in Tor

Tor relays are run by **volunteers**. So they can be **malicious**.



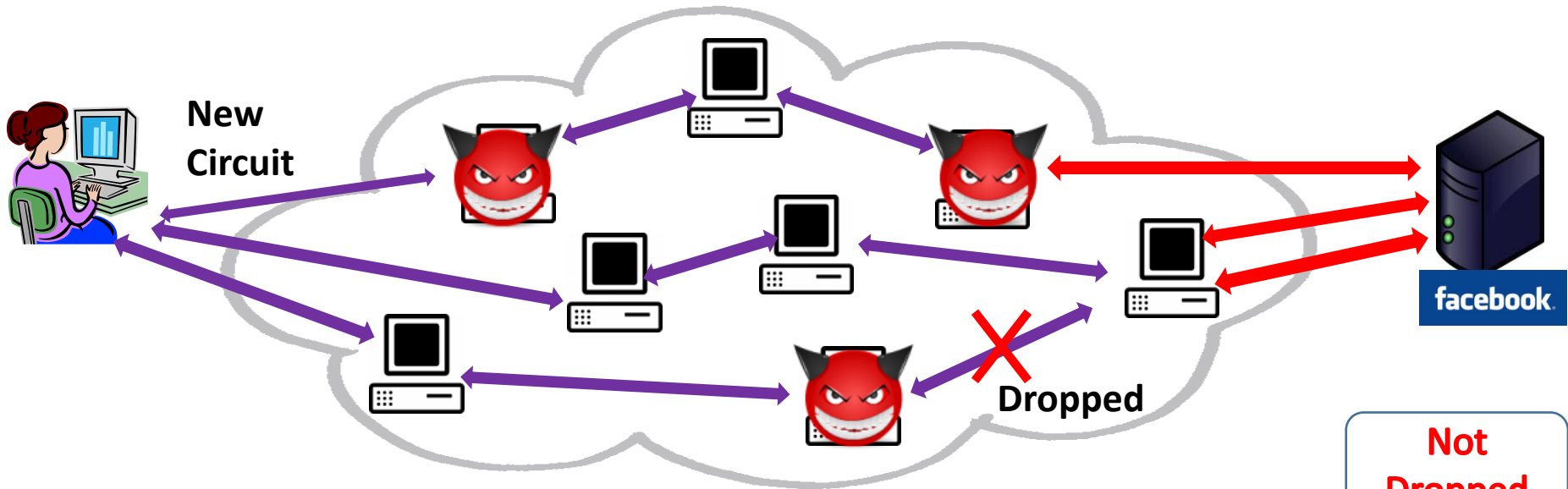
Assuming **c** fraction of the bandwidth is controlled by an adversary

Probability of circuits being **compromised**:

$$\Pr(\text{compromised}) = c^2$$



Selective DoS in Tor



C- Compromised
H- Honest Relay

Dropped

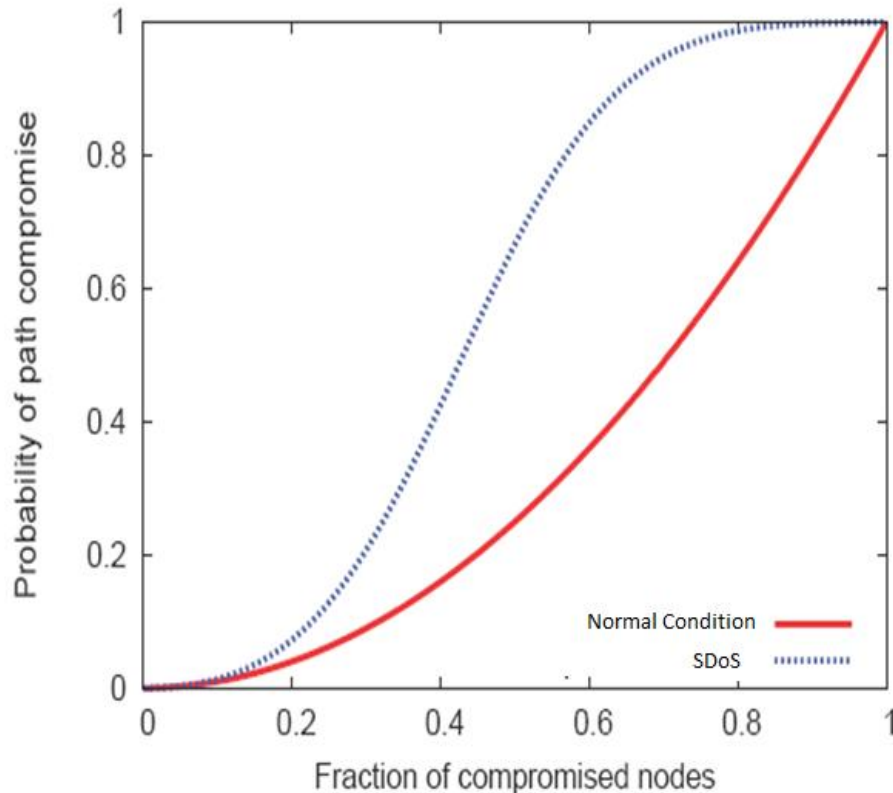
Entry	Middle	Exit
H	H	C
H	C	C
H	C	H
C	H	H
C	C	H

Entry	Middle	Exit
C	H	C
C	C	C
H	H	H

**Not
Dropped**



Impact of Selective DoS



Under Normal Condition:

$$\Pr(\text{Compromised}) = c^2$$

Under Selective DoS:

Entry	Middle	Exit
C	H	C
C	C	C
H	H	H

$$\Pr(\text{Compromised}) = \frac{c^2}{c^2 + (1-c)^3}$$

For a given value of c , $\frac{c^2}{c^2 + (1-c)^3} > c^2$



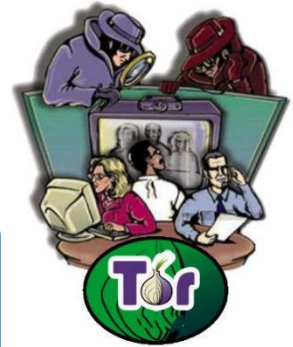
Recent Attacks on Tor

U.S. NEWS

Federal Agents Pierce Tor Web-Anonymity Tool

Child-Pornography Investigation Gets Around Browser Designed to Mask Identities

Some law-enforcement officers and researchers say the shakiness of the network itself, which relies on volunteers, presents opportunities for authorities to trace users



GIZMODO

Search

The NSA's Been Trying to Hack into Tor's Anonymous Internet For Years



Adam Clark Estes

Filed to: NSA 10/04/13 11:53am

25,057 🔥 ★ ▼

One method involves tapping into the Tor nodes and look for patterns in people entering and leaving the network

OCTOBER 22, 2012 | BY EVA GALPERIN AND KATITZA RODRIGUEZ AND PETER ECKERSLEY



Dutch Government Proposes Cyberattacks Against... Everyone

It would allow Dutch police to use exploits and malware against privacy systems like the Tor network



Our Goal

We outline the following goals:

1. Capture the dropping characteristics of compromised relays through a localized reputation framework.
2. Enable the reputation metric to penalize frequent oscillating behavior.
3. Use a filtering protocol to filter out potentially compromised relay.

Threat Model:

1. Small fraction ($\sim 20\%$) of relays are compromised.
2. Compromised relays drop non-compromised circuits at different rates.



PID Controller

We propose leveraging a PID controller-based reputation metric to handle dynamic behavioral change by malicious relays.

The classical PID controller-based update function:

$$u(t) = \alpha \cdot e(t) + \beta \cdot \underbrace{\frac{1}{t} \int_0^t e(x) dx}_{\text{Aggregation of Past Feedback}} + \gamma \cdot e'(t)$$

Current Feedback

Sudden Fluctuation

Aggregation of Past Feedback



Reputation Metric

We use the following discrete version of a PID controller-based update function as our reputation metric:

$$R_n[i] = \alpha \cdot R[i] + \beta \cdot H[i] + \gamma(D[i]) \cdot D[i]$$

Current Feedback

$$R[i] = \begin{cases} 0.01, & \text{circuit failed} \\ 1, & \text{otherwise} \end{cases}$$

Historical Feedback

$$H[i] = \frac{\sum_{k=1}^{maxH} R[i-k] \cdot \omega_k}{\sum_{k=1}^{maxH} \omega_k}$$

Sudden Fluctuation

$$D[i] = R[i] - H[i]$$
$$\gamma(x) = \begin{cases} \gamma_1, & x \geq 0 \\ \gamma_2, & x < 0 \end{cases}$$



Reputation Metric Parameters

PID controller-based reputation metric:

$$R_n[i] = \alpha \cdot R[i] + \beta \cdot H[i] + \gamma(D[i]) \cdot D[i]$$

Parameters:

α : Defines the contribution of recent performance

β : Defines the contribution of past performance

γ : Defines the contribution of sudden change in performance

We determine the value (or relationship) of these parameters so that the reputation metric can effectively capture **behavioral oscillation**.



Setting Parameters

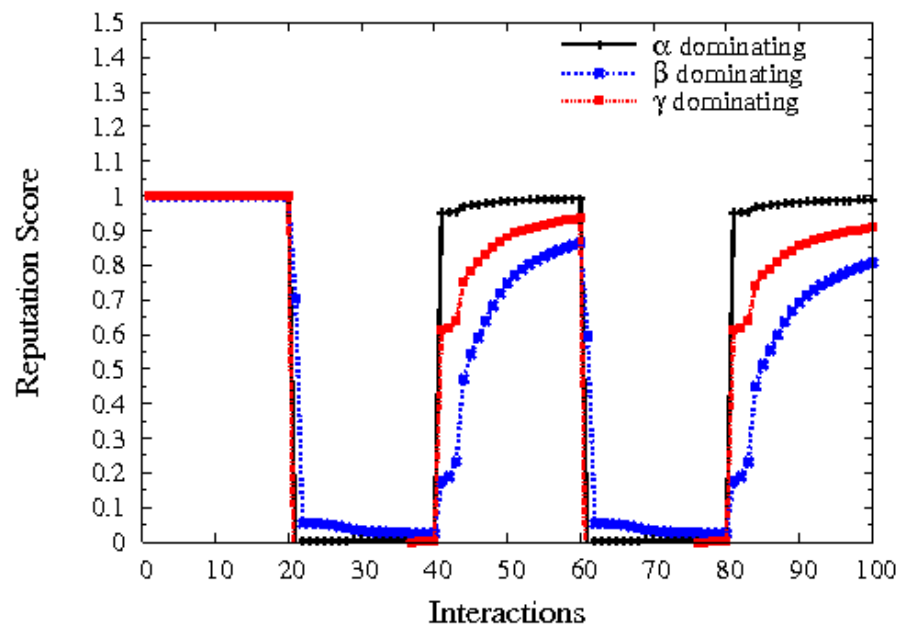
To better understand the relationship between the parameters we analyze the following three cases.

Cases:

1. $\alpha \gg \beta, \gamma$
2. $\beta \gg \alpha, \gamma$
3. $\gamma \gg \alpha, \beta$

In our setting we
select: $\beta \gg \alpha, \gamma$ &
 $\gamma_2 > \gamma_1$

Alter behavior every 20 interactions



To penalize oscillating behavior we give higher weight to historical performance. We also give higher weight to sudden drop in performance than sudden rise in performance.



Filtering Protocol

We propose a simple mean and standard deviation based filtering protocol to filter compromised relays.

The filtering technique employs the following steps:

1. First compute the mean (μ) and standard deviation (σ) of the top **(1-c)%** of the relays, where **c** represents the expected fraction of malicious relays.
2. Filter any relay whose reputation score lies outside the range: $(\mu - k\sigma, \mu + k\sigma)$, where **k** represents the degree of deviation from the expected reputation score.

For guards we keep only the **highest reputable** relay.



Evaluation

We evaluate our approach through simulation.

Simulation setup:

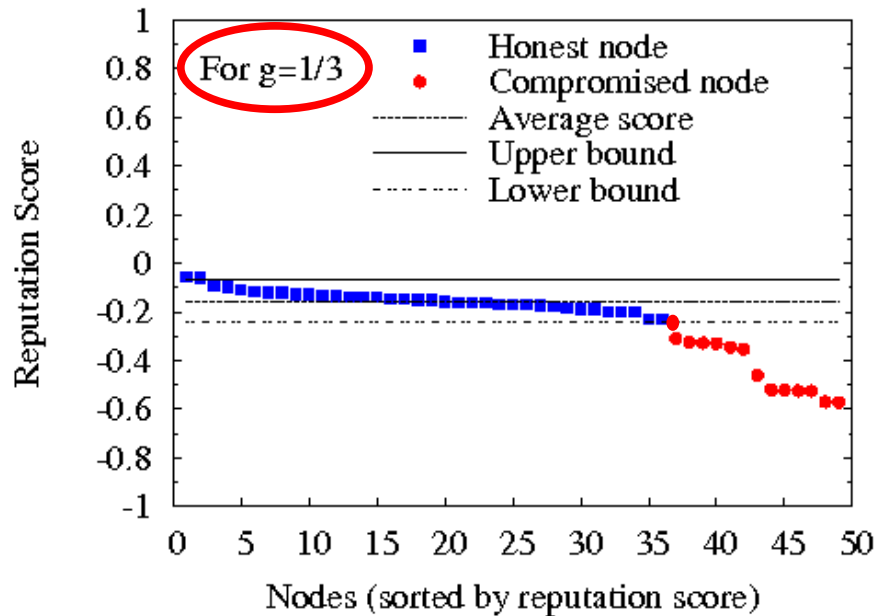
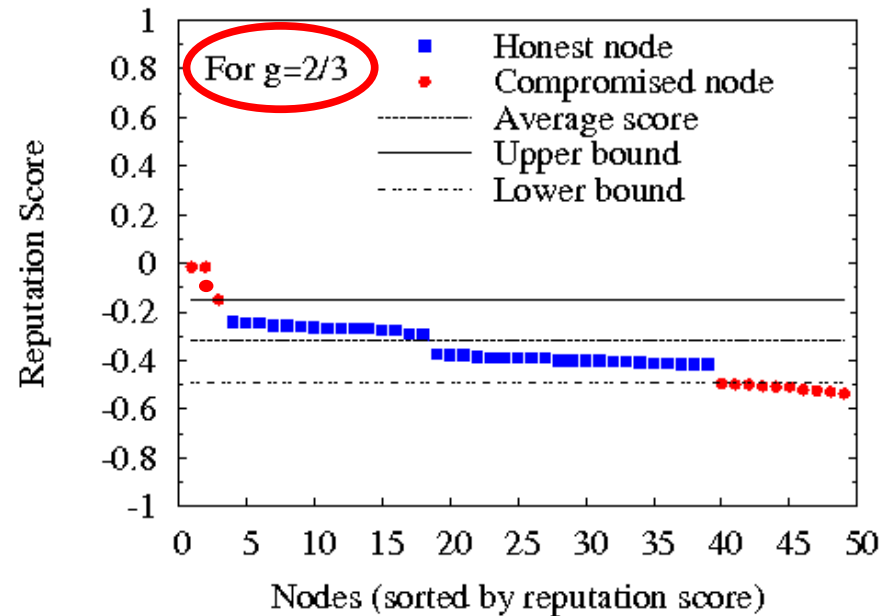
- Gather live Tor relay information (IP, Bandwidth, Selection probabilities) from <https://compass.torproject.org/>
- Randomly assigned **20%** bandwidth to be compromised.

	Parameter	Description	Value/Range
Computational	α	Proportional gain	0.1
	β	Integral gain	0.9
	γ_1	Positive derivative gain	0.05
	γ_2	Negative derivative gain	0.2
Environmental	g	Fraction of malicious guard	{0,1/3,2/3,1}
	d	Drop rate by malicious relays	[0,1]
	f	Transient Network failure	0.21

Transient network failure computed using *TorFlow* project <https://gitweb.torproject.org/torflow.git>



Filtering Malicious Relays



1. With $g=1/3$ we can filter malicious relays efficiently.
2. Even with $g=2/3$ we can filter significant portion of the malicious relays.

Majority of the relays are honest (80%)
So μ and σ are more influenced by honest relays

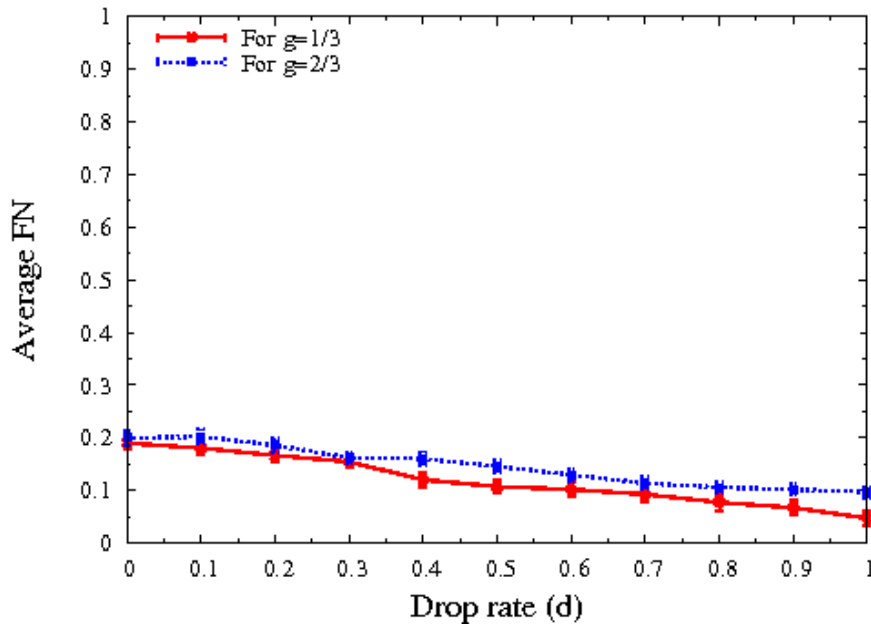
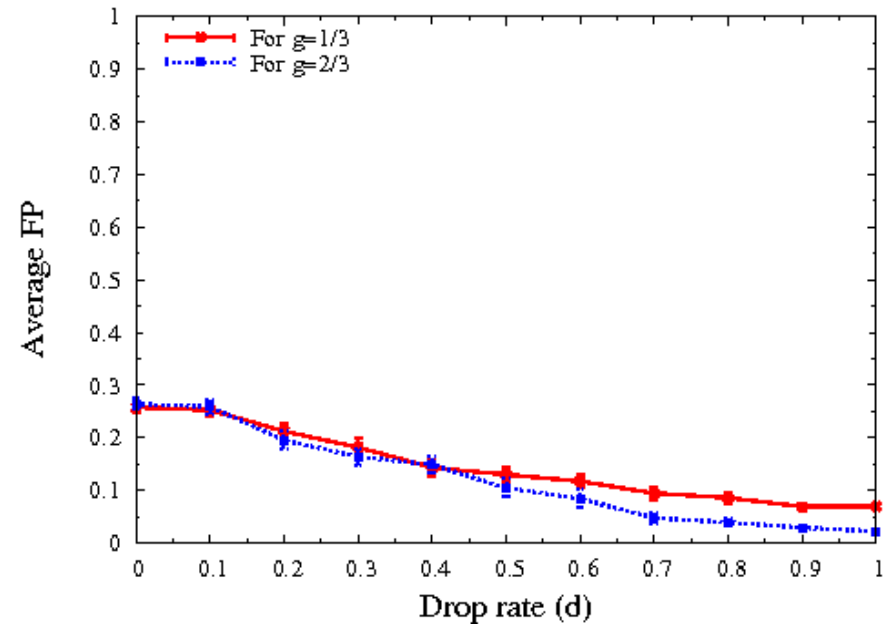


False Errors

False Negative (**FN**)= Fraction of compromised relays accepted

False Positive (**FP**)= Fraction of honest relays discarded

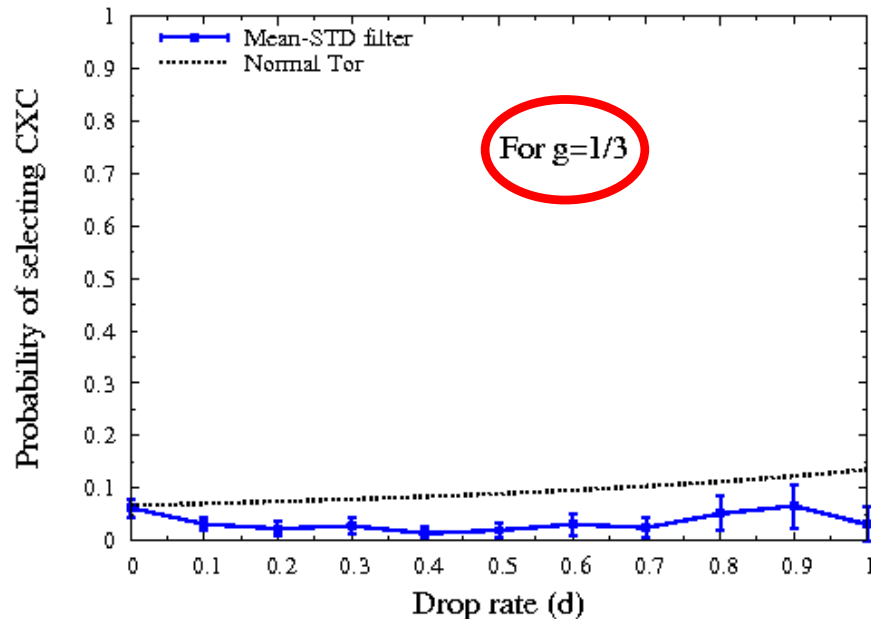
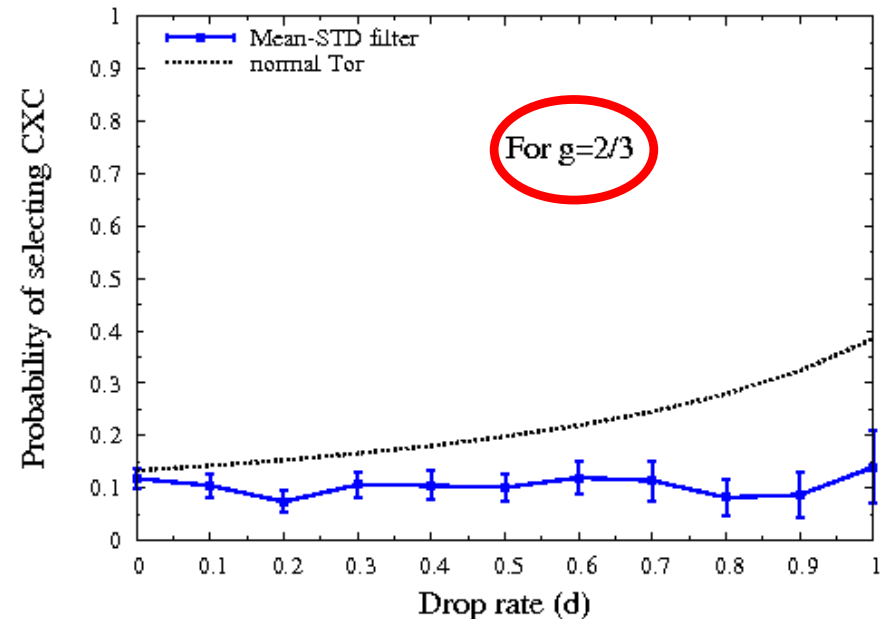
Ideally we want both **FN** and **FP** as small as possible.



As drop rate d increases both FN and FP decrease.



Selecting Compromised Circuits



Probability of selecting a compromised circuit after filtering:

$$\Pr(CXC) = \frac{g_f \cdot c_f}{g_f \cdot c_f + (1 - g_f)(1 - c_f)^2 + (1 - d)[1 - g_f \cdot c_f - (1 - g_f)(1 - c_f)^2]}$$

Compared to conventional Tor our approach provides better resiliency



Limitations and Future Work

1. In the absence of attacks, a small fraction of honest relays are classified as outliers due to random network failures.

Filtered relays may represent congested relays in absence of attack

2. We only considered a specific type of circuit dropping, namely, selective DoS and its probabilistic variants.

Other forms of attack like **targeted attack** exist

3. New users benefit from our reputation model only after a certain amount of usage.



Conclusion

Our work shows:

1. PID controller-based reputation framework can successfully capture the oscillating dropping characteristics of compromised relays.
2. Our local reputation framework coupled with our filtering protocol was able to successfully filter out compromised relays.

As an extension of this work we have analyzed more complex attack scenarios against our reputation framework. (It is to appear at ASIACCS 2014)



End

