

Theory and Design of Low-latency Anonymity Systems (Lecture 3)

Paul Syverson

U.S. Naval Research Laboratory

syverson@itd.nrl.navy.mil

<http://www.syverson.org>



Reminders

Paul, don't trail off in volume when speaking.

Attendees, if Paul trails off and you want to hear what he says, speak up.

On question someone asked after end of Lecture 2 about JavaScript in long-path congestion attacks

- Yes, blocking all JavaScript will block attacks used to generate data in paper.
- But blocking all JavaScript blocks much of the web,
 - Most anonymity tools sanitize but don't block outright.
- Other bases available for attack: HTTP header refresh, or HTML with embedded tiny images.

Course Outline

Lecture 1:

- Usage examples, basic notions of anonymity, types of anonymous comms systems
- Crowds: Probabilistic anonymity, predecessor attacks

Lecture 2:

- Onion routing basics: simple demo of using Tor, network discovery, circuit construction, crypto, node types and exit policies
- Economics, incentives, usability, network effects

Course Outline

Lecture 3:

- Formalization and analysis, possibilistic and probabilistic definitions of anonymity
- Hidden services: responder anonymity, predecessor attacks revisited, guard nodes

Lecture 4:

- Link attacks
- Trust

Formal analysis of onion routing

Possibilistic characterization using IO automata

Probabilistic analysis abstracting IO automata
characterization to a black box

Anonymous Communication

	Deployed	Analyzed
Mix Networks	+	+
Dining cryptographers	-	+
Onion routing	+	-
Crowds	-	+

Possibilistic Analysis Overview

Formally model onion routing using input/output automata

Characterize the situations that provide anonymity

Possibilistic Analysis Overview

Formally model onion routing using input/output automata

- Simplified onion-routing protocol

- Non-cryptographic analysis

Characterize the situations that provide anonymity

Possibilistic Analysis Overview

Formally model onion routing using input/output automata

- Simplified onion-routing protocol

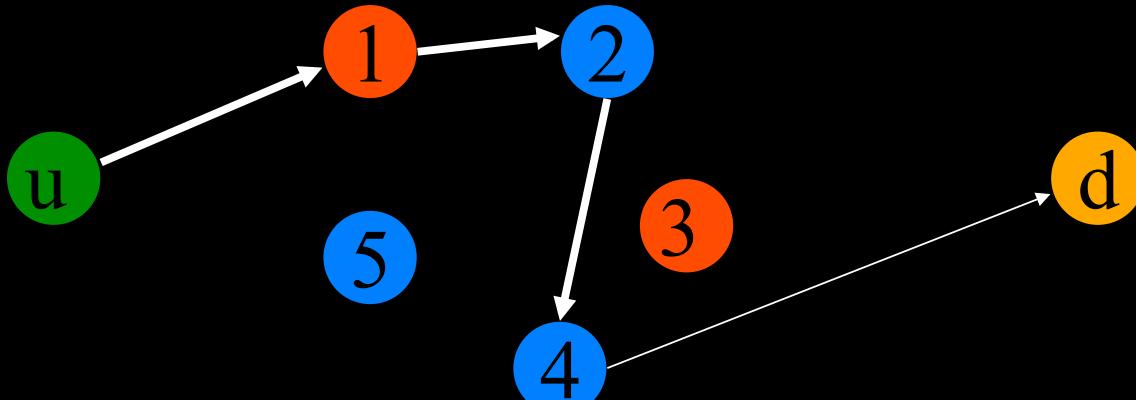
- Non-cryptographic analysis

Characterize the situations that provide anonymity

- Send a message, receive a message, communicate with a destination

- Possibilistic anonymity

Main Theorem



Main theorem: Adversary can only determine the parts of a circuit it controls or is next to.

Anonymous Communication

Sender anonymity: Adversary can't determine the sender of a given message

Receiver anonymity: Adversary can't determine the receiver of a given message

Unlinkability: Adversary can't determine who talks to whom

Model

Constructed with I/O automata

Models asynchrony

Relies on abstract properties of cryptosystem

Simplified onion-routing protocol

No key distribution

No circuit teardowns

No separate destinations

No streams

No stream cipher

Each user constructs a circuit to one destination

Circuit identifiers

Input/Ouput Automata

States

Actions

Input, ouput, internal

Actions transition between states

Every state has *enabled* actions

Input actions are always enabled

Alternating state/action sequence is an *execution*

In *fair* executions actions enabled infinitely often occur infinitely often

In *cryptographic* executions no encrypted control messages are sent before they are received unless the sender possesses the key

I/O Automata Model

Automata

User

Server

Fully-connected network of
FIFO Channels

Adversary replaces some
servers with arbitrary automata

Notation

U is the set of users

R is the set of routers

$N = U \cup R$ is the set of all
agents

$A \subseteq N$ is the adversary

K is the keyspace

l is the (fixed) circuit length

$k(u, c, i)$ denotes the i th key
used by user u on circuit c

User automaton

```

1:  $c \in \{(r_1, \dots, r_l) \in R^l \mid \forall_i r_i \neq r_{i+1}\}$ ; init: arbitrary           ▷ User's circuit
2:  $i \in \mathbf{N}$ ; init: random                                         ▷ Circuit identifier
3:  $b \in \mathbf{N}$ ; init: 0                                              ▷ Next hop to build

4: procedure START
5:   SEND( $c_1, [i, 0, \{\text{CREATE}\}_{k(u,c,1)}]$ )
6:    $b = 1$ 
7: end procedure
8: procedure MESSAGE( $msg, j$ )                                     ▷  $msg \in M$  received from  $j \in N$ 
9:   if  $j = c_1$  then
10:    if  $b = 1$  then
11:      if  $msg = [i, 0, \text{CREATED}]$  then
12:         $b++$ 
13:        SEND( $c_1, [i, 0, \{[\text{EXTEND}, c_b, \{\text{CREATE}\}_{k(u,c,b)}]\}_{k(u,c,b-1), \dots, k(u,c,1)}]$ )
14:      end if
15:    else if  $b < l$  then
16:      if  $msg = [i, 0, \{\text{EXTENDED}\}_{k(u,c,b-1), \dots, k(u,c,1)}]$  then
17:         $b++$ 
18:        SEND( $c_1, [i, 0, \{[\text{EXTEND}, c_b, \{\text{CREATE}\}_{k(u,c,b)}]\}_{k(u,c,b-1), \dots, k(u,c,1)}]$ )
19:      end if
20:    else if  $b = l$  then
21:      if  $msg = [i, 0, \{\text{EXTENDED}\}_{k(u,c,b-1), \dots, k(u,c,1)}]$  then
22:         $b++$ 
23:      end if
24:    end if
25:  end if
26: end procedure

```

Server automaton

```

1:  $keys \subseteq K$ , where  $|keys| \geq |U| \cdot \lceil \frac{l}{2} \rceil$ ; init: arbitrary           ▷ Private keys
2:  $T \subset N \times \mathbf{N} \times R \times \mathbf{Z} \times keys$ ; init:  $\emptyset$                       ▷ Routing table
3: procedure MESSAGE( $[i, n, p], q$ )          ▷  $[i, n, p] \in M$  received from  $q \in N$ 
4:   if  $[q, n, \emptyset, -1, k] \in T$  then          ▷ In link created, out link absent
5:     if  $\exists_{s \in R-r, b \in P} p = \{\text{EXTEND}, s, b\}_k$  then
6:       SEND( $s, [minid(T, s), b]$ )
7:        $T = T - [q, n, \emptyset, -1, k] + [q, n, s, -minid(T, s), k]$ 
8:     end if
9:   else if  $[s, m, q, -n, k] \in T$  then          ▷ In link created, out link initiated
10:    if  $p = \text{CREATED}$  then
11:       $T = T - [s, m, q, -n, k] + [s, m, q, n, k]$ 
12:      SEND( $s, [i, m, \{\text{EXTENDED}\}_k]$ )
13:    end if
14:   else if  $\exists_{m>0} [q, n, s, m, k] \in T$  then          ▷ In and out links created
15:     SEND( $s, [i, m, \{p\}_{-k}]$ )
16:   else if  $[s, m, q, n, k] \in T$  then          ▷ Forward message down the circuit
17:     SEND( $s, [i, m, \{p\}_k]$ )                  ▷ In and out links created
18:   else
19:     if  $\exists_{k \in keys} p = \{\text{CREATE}\}_k$  then          ▷ Forward message up the circuit
20:        $T = T + [q, n, \emptyset, -1, k]$ 
21:       SEND( $q, [i, n, \text{CREATED}]$ )
22:     end if
23:   end if
24: end procedure

```

Anonymity

Definition configuration

A **configuration** is a function $U \rightarrow R^l$ mapping each user to his circuit.

Anonymity

Definition configuration

configuration is a function $U \rightarrow R^l$ mapping each user to his circuit.

Definition (indistinguishability):

Executions α and β are **indistinguishable** to adversary A when his actions in β are the same as in α after possibly applying the following:

ξ : A permutation on the keys not held by A .

π : A permutation on the messages encrypted by a key not held by A .

Anonymity

Definition (anonymity):

User u performs action α ***anonymously*** in *configuration* C with respect to adversary A if, for every execution of C in which u performs α , there exists an execution that is *indistinguishable* to A in which u does not perform α .

Anonymity

Definition (anonymity):

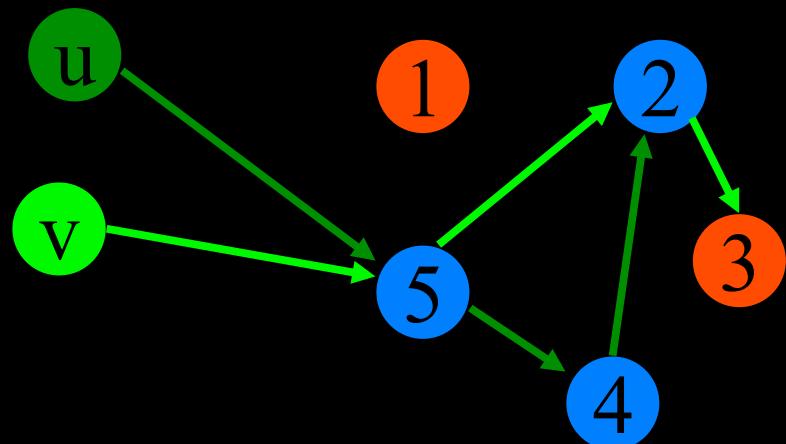
User u performs action α ***anonymously*** in *configuration* C with respect to adversary A if, for every execution of C in which u performs α , there exists an execution that is *indistinguishable* to A in which u does not perform α .

Definition unlinkability

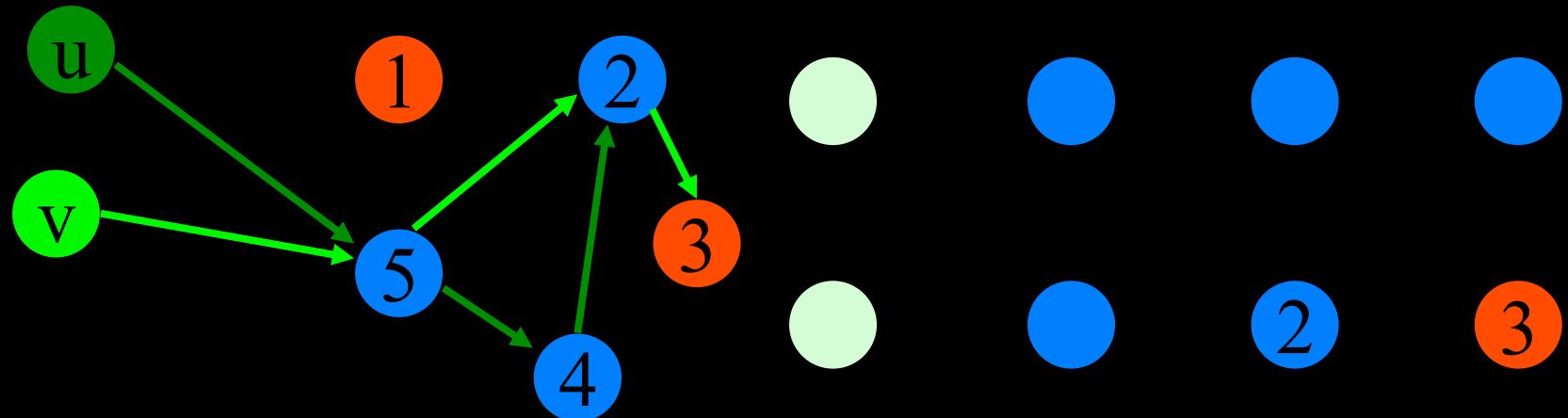
User u is ***unlinkable*** to d in configuration C with respect to adversary A if, for every fair, cryptographic execution of C in which u talks to d , there exists a fair, cryptographic execution that is indistinguishable to A in which u does not talk to d .

Theorem: Let C and D be configurations for which there exists a permutation $\rho: U \rightarrow U$ such that $C_i(u) = D_i(\rho(u))$ if $C_i(u)$ or $D_i(\rho(u))$ is compromised or is adjacent to a compromised router. Then for every fair, cryptographic execution α of C there exists an indistinguishable, fair, cryptographic execution β of D . The converse also holds.

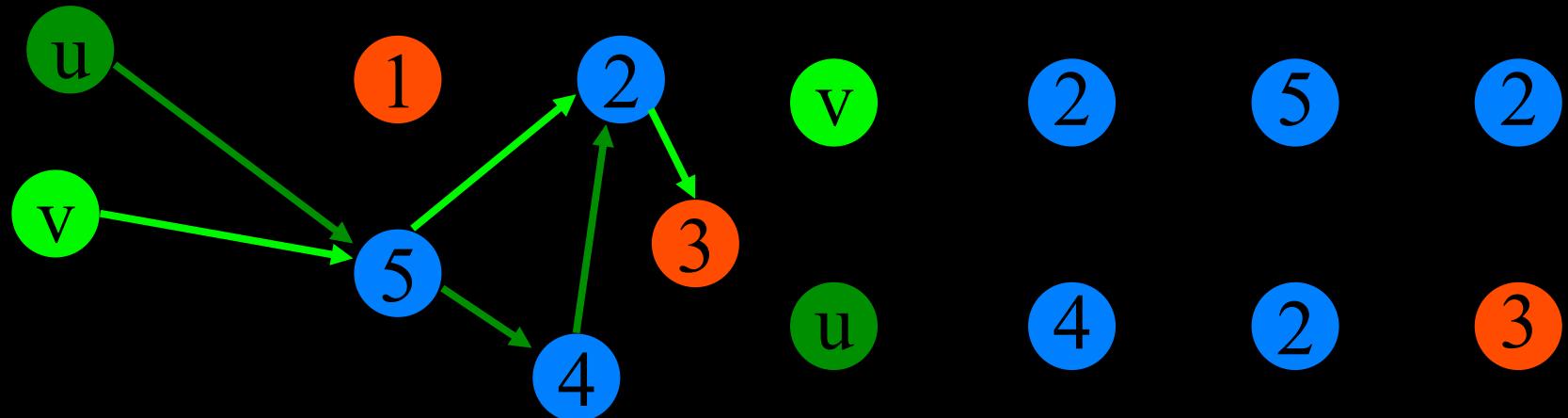
Theorem: Let C and D be configurations for which there exists a permutation $\rho: U \rightarrow U$ such that $C_i(u) = D_i(\rho(u))$ if $C_i(u)$ or $D_i(\rho(u))$ is compromised or is adjacent to a compromised router. Then for every fair, cryptographic execution α of C there exists an indistinguishable, fair, cryptographic execution β of D . The converse also holds.



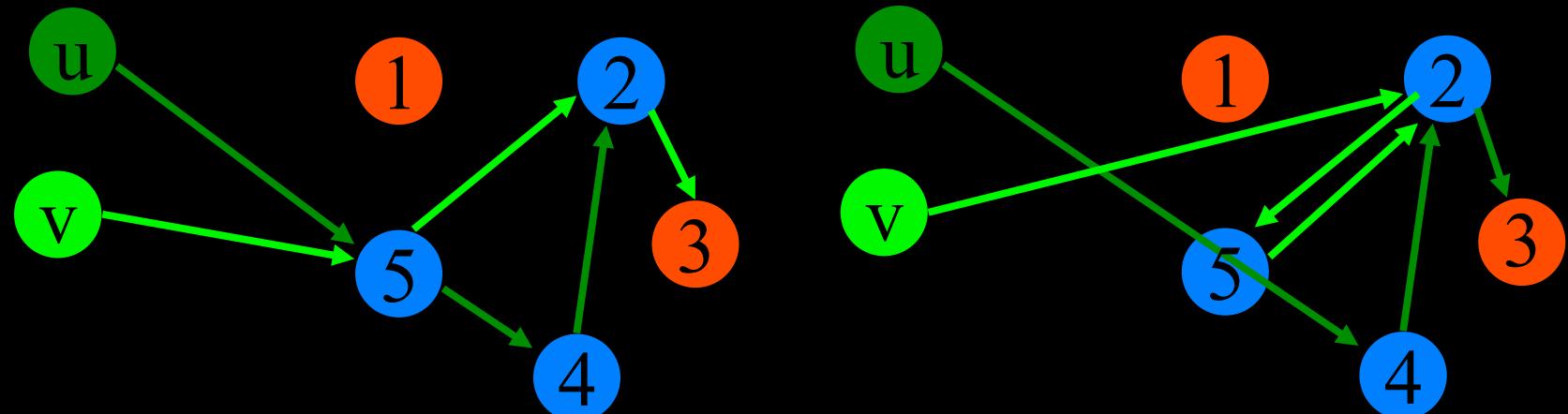
Theorem: Let C and D be configurations for which there exists a permutation $\rho: U \rightarrow U$ such that $C_i(u) = D_i(\rho(u))$ if $C_i(u)$ or $D_i(\rho(u))$ is compromised or is adjacent to a compromised router. Then for every fair, cryptographic execution α of C there exists an indistinguishable, fair, cryptographic execution β of D . The converse also holds.



Theorem: Let C and D be configurations for which there exists a permutation $\rho: U \rightarrow U$ such that $C_i(u) = D_i(\rho(u))$ if $C_i(u)$ or $D_i(\rho(u))$ is compromised or is adjacent to a compromised router. Then for every fair, cryptographic execution α of C there exists an indistinguishable, fair, cryptographic execution β of D . The converse also holds.



Theorem: Let C and D be configurations for which there exists a permutation $\rho: U \rightarrow U$ such that $C_i(u) = D_i(\rho(u))$ if $C_i(u)$ or $D_i(\rho(u))$ is compromised or is adjacent to a compromised router. Then for every fair, cryptographic execution α of C there exists an indistinguishable, fair, cryptographic execution β of D . The converse also holds.



Unlinkability

Corollary: A user is unlinkable to its destination when:

Unlinkability

Corollary: A user is unlinkable to its destination when:

u

3

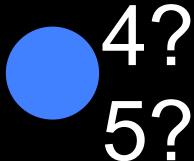
2

4?
5?

The last router is
unknown.

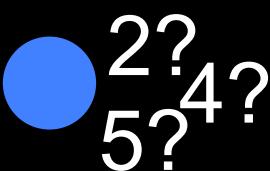
Unlinkability

Corollary: A user is unlinkable to its destination when:



The last router is unknown.

OR



The user is unknown and another unknown user has an unknown destination.

Unlinkability

Corollary: A user is unlinkable to its destination when:

u

3

2

4?
5?

The last router is unknown.

OR

2

1

4

5

2?
4?
5?

The user is unknown and another unknown user has an unknown destination.

OR

2

1

4

5

1

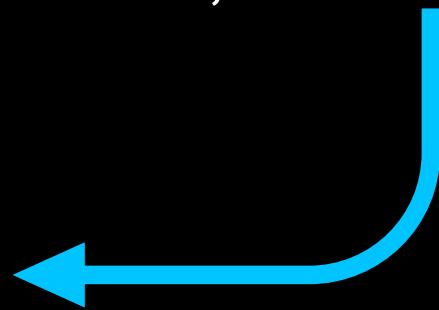
2

The user is unknown and another unknown user has a different destination.

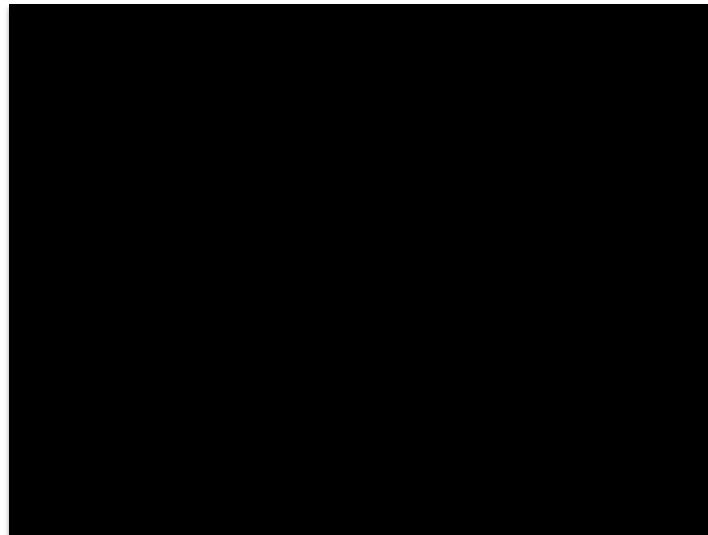
Probabilistic anonymity

Possibilistic result is nice, but we would like to quantify the anonymity provided by a system

And we want to use a black box model, like this



Probabilistic Analysis of Onion Routing in a Black-box Model



In this portion we will

1. Use a black-box abstraction to create a probabilistic model of onion routing
2. Analyze unlinkability
 - a. Provide worst-case bounds
 - b. Examine a typical case

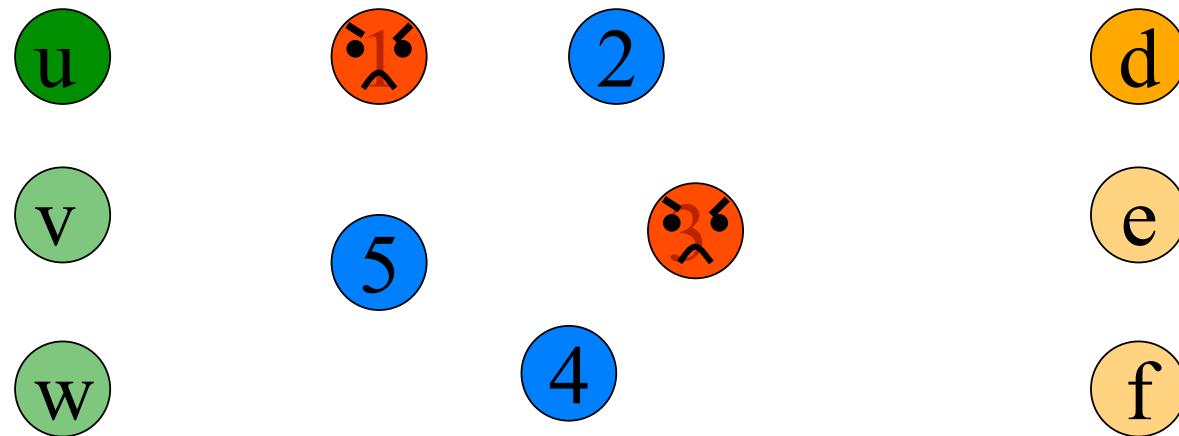
Anonymous Communication

- *Sender anonymity*: Adversary can't determine the sender of a given message
- *Receiver anonymity*: Adversary can't determine the receiver of a given message
- *Unlinkability*: Adversary can't determine who talks to whom

Anonymous Communication

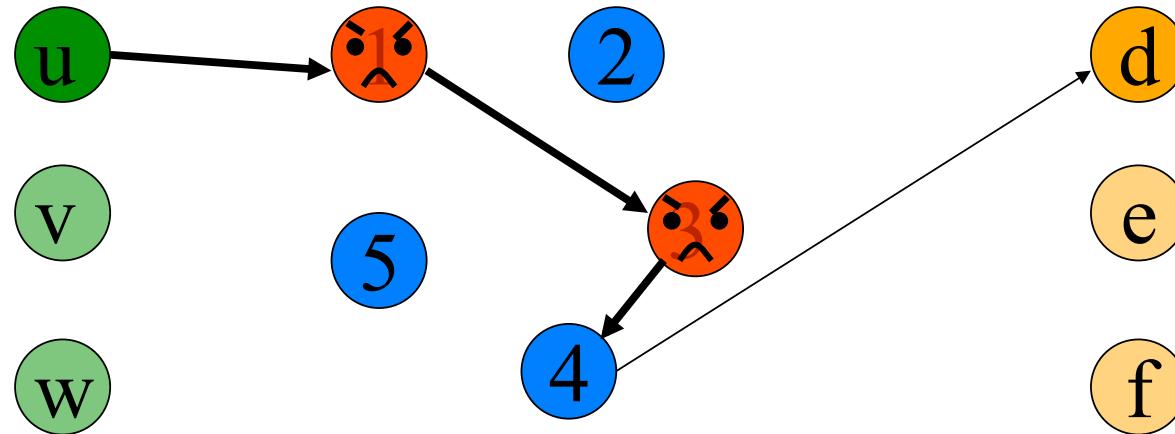
- *Sender anonymity*: Adversary can't determine the sender of a given message
- *Receiver anonymity*: Adversary can't determine the receiver of a given message
- *Unlinkability*: Adversary can't determine who talks to whom

Adversary positions on circuits



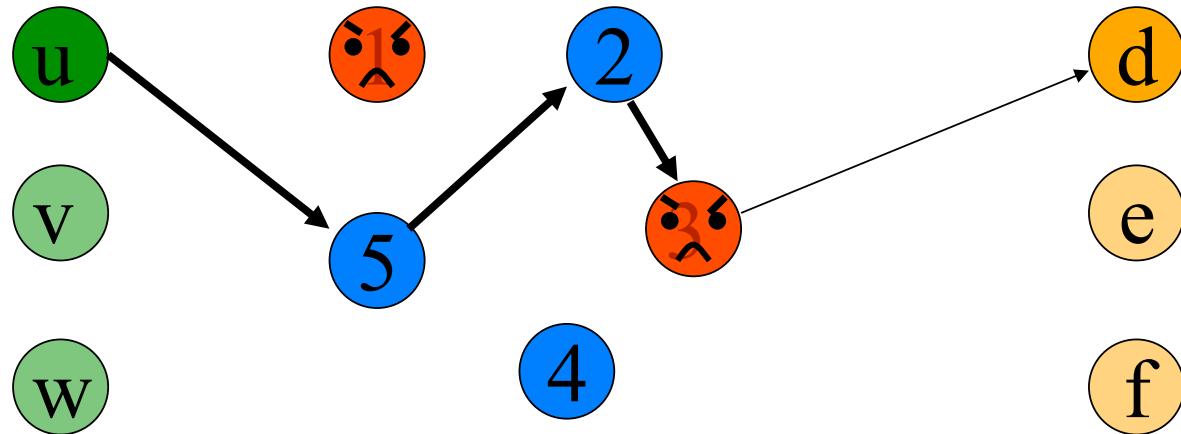
- 1.
- 2.
- 3.
- 4.

Adversary positions on circuits



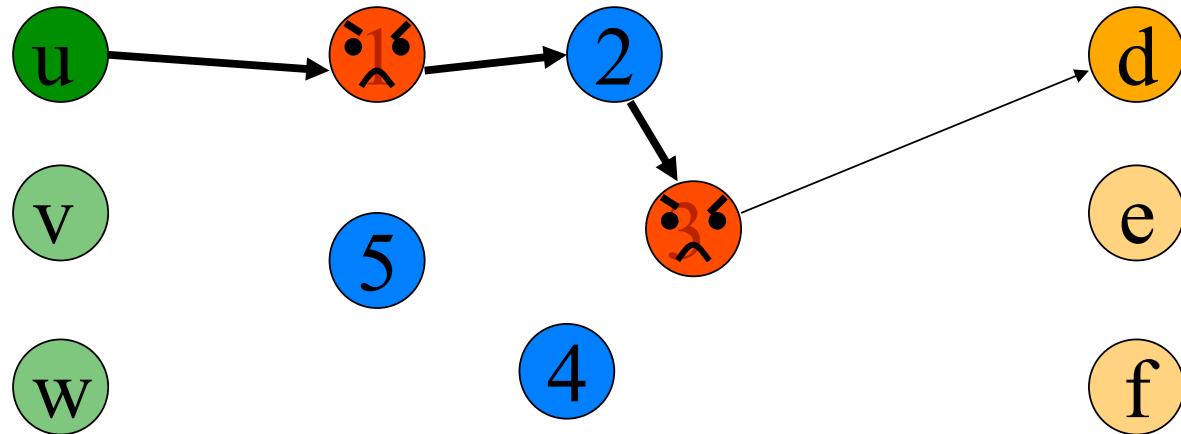
1. First router compromised
- 2.
- 3.
- 4.

Adversary positions on circuits



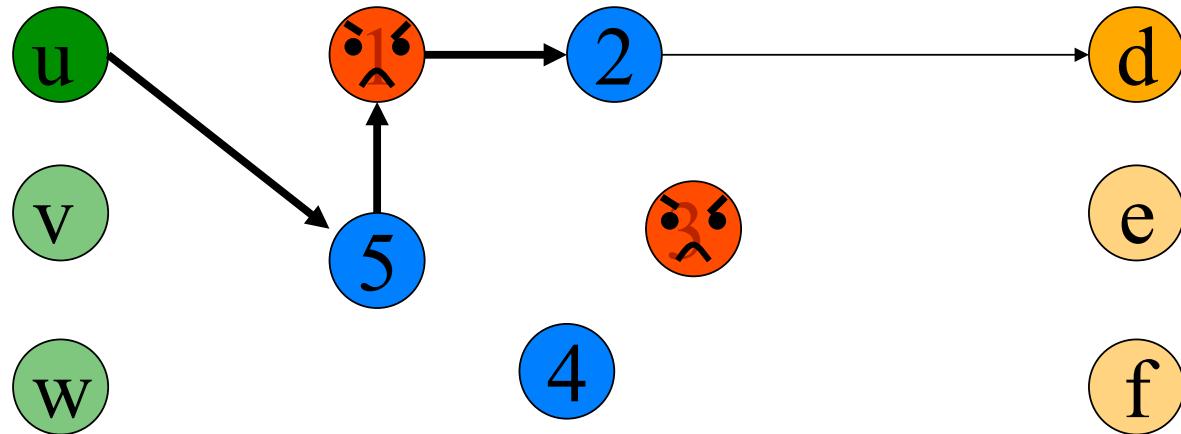
1. First router compromised
2. Last router compromised
- 3.
- 4.

Adversary positions on circuits



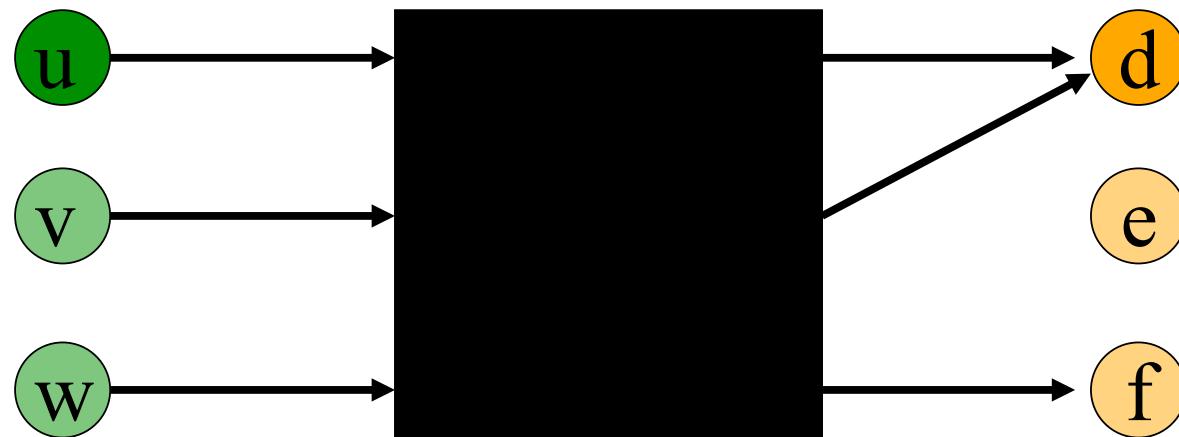
1. First router compromised
2. Last router compromised
3. First and last compromised
- 4.

Adversary positions on circuits

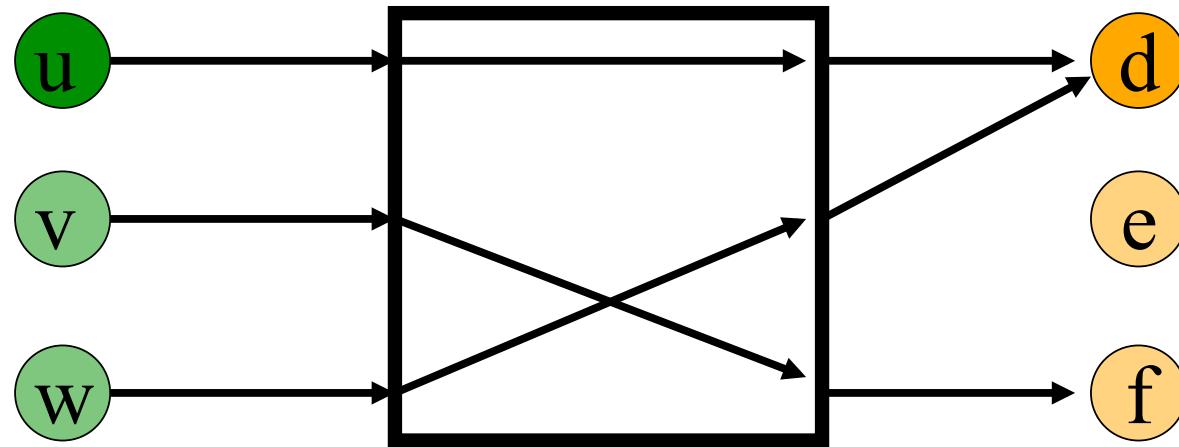


1. First router compromised
2. Last router compromised
3. First and last compromised
4. Neither first nor last compromised

Black-box Abstraction

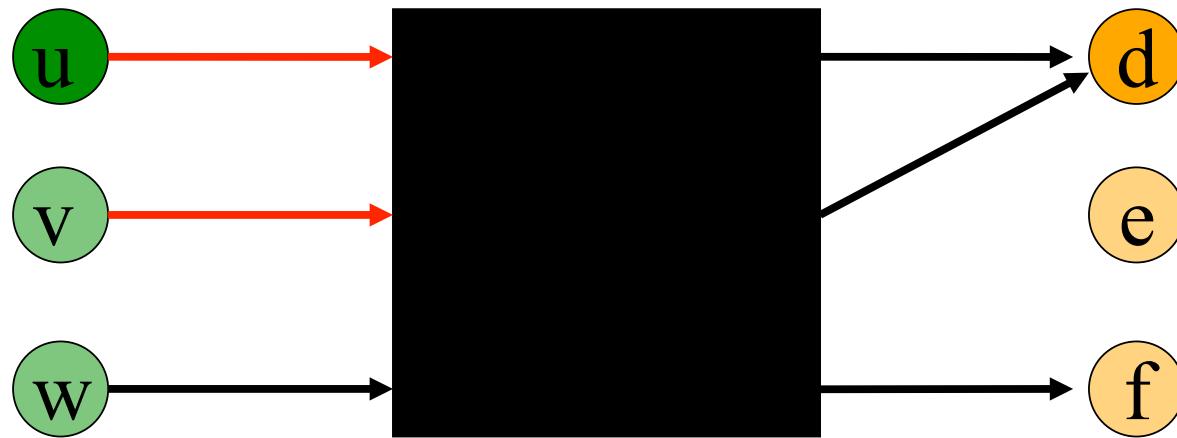


Black-box Abstraction



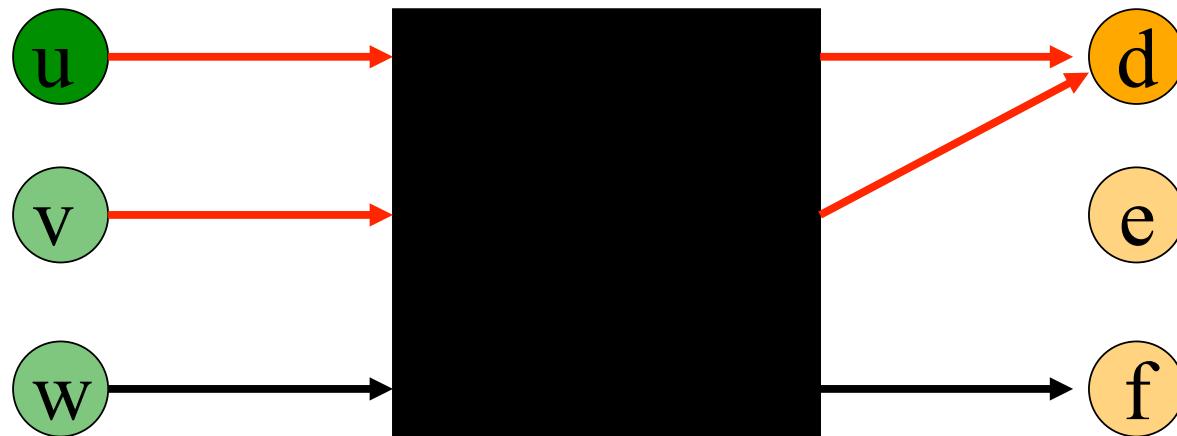
1. Users choose a destination

Black-box Abstraction



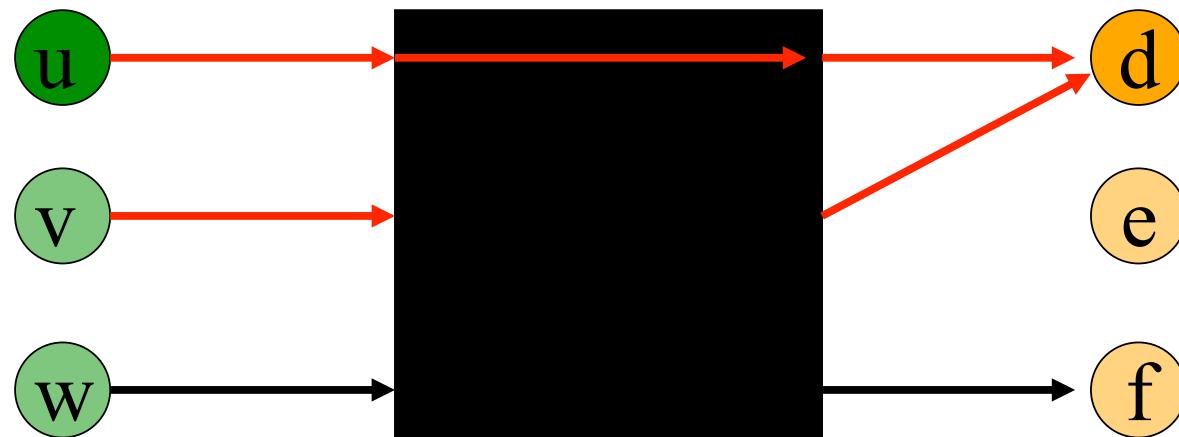
1. Users choose a destination
2. Some inputs are observed

Black-box Abstraction



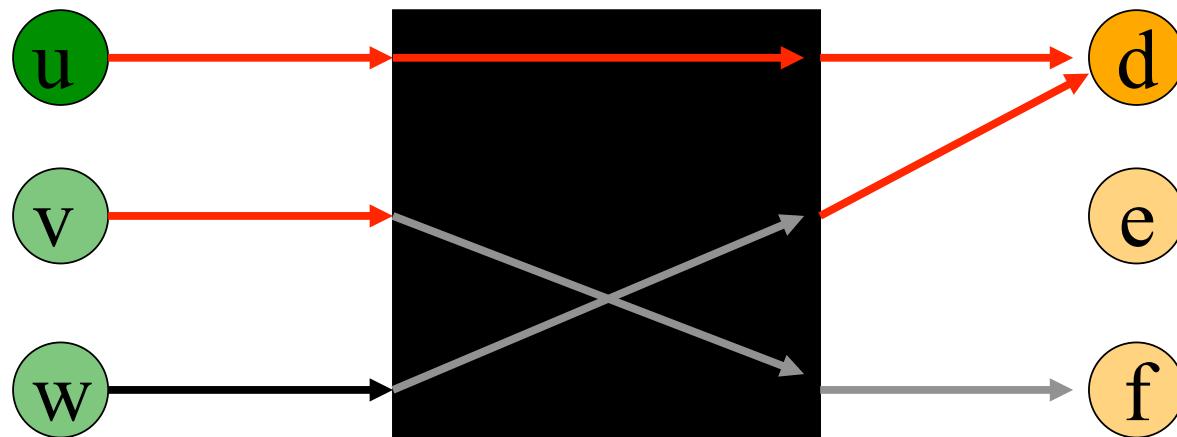
1. Users choose a destination
2. Some inputs are observed
3. Some outputs are observed

Black-box Anonymity



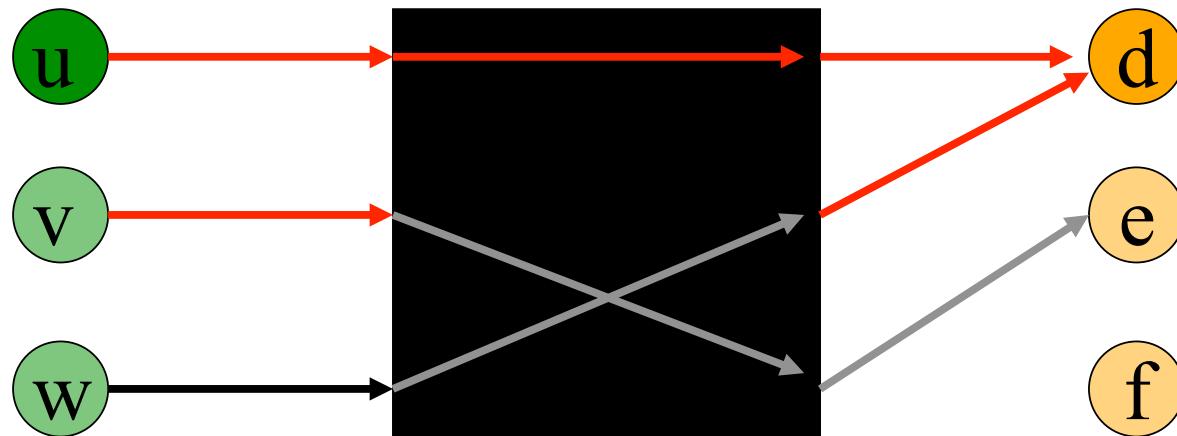
- The adversary can link observed inputs and outputs of the same user.

Black-box Anonymity



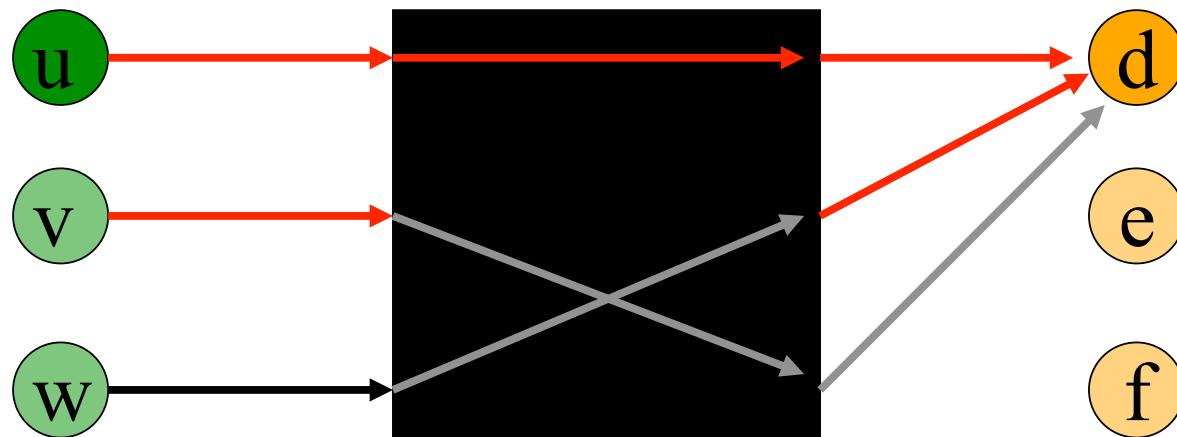
- The adversary can link observed inputs and outputs of the same user.
- Any configuration consistent with these observations is indistinguishable to the adversary.

Black-box Anonymity



- The adversary can link observed inputs and outputs of the same user.
- Any configuration consistent with these observations is indistinguishable to the adversary.

Black-box Anonymity



- The adversary can link observed inputs and outputs of the same user.
- Any configuration consistent with these observations is indistinguishable to the adversary.

Probabilistic Black-box

u

v

w



d

e

f

Probabilistic Black-box

u

v

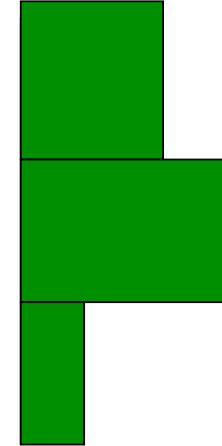
w



d

e

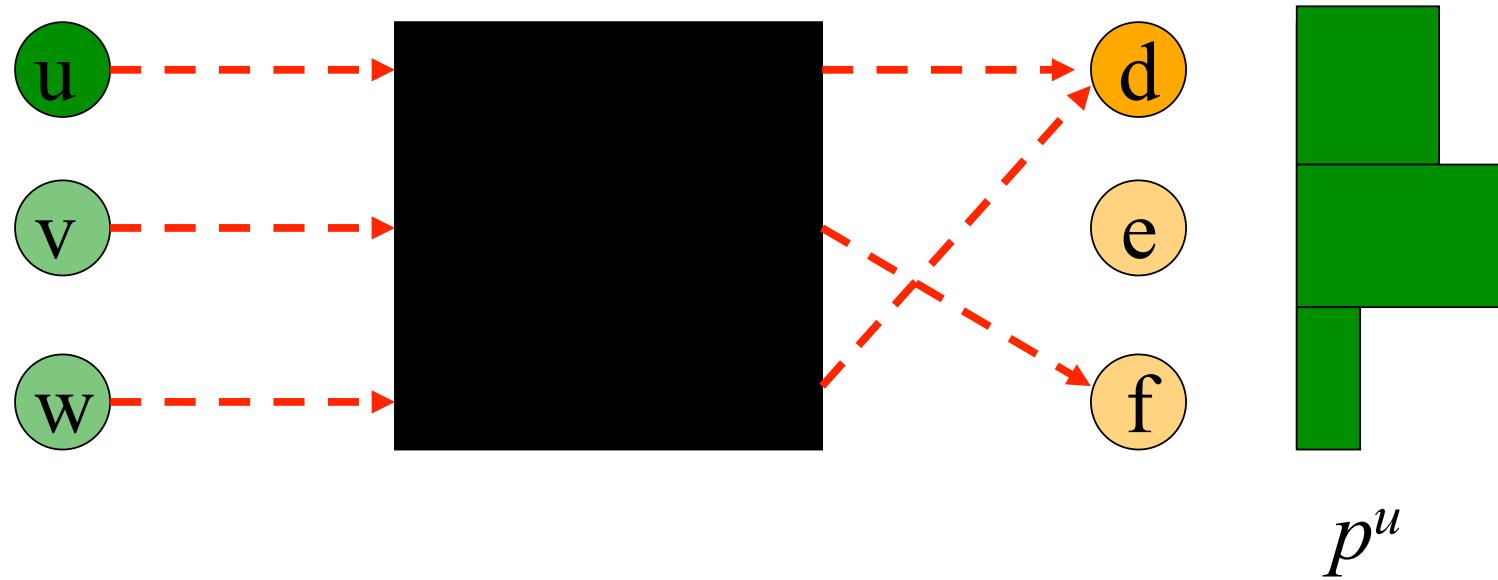
f



$$p^u$$

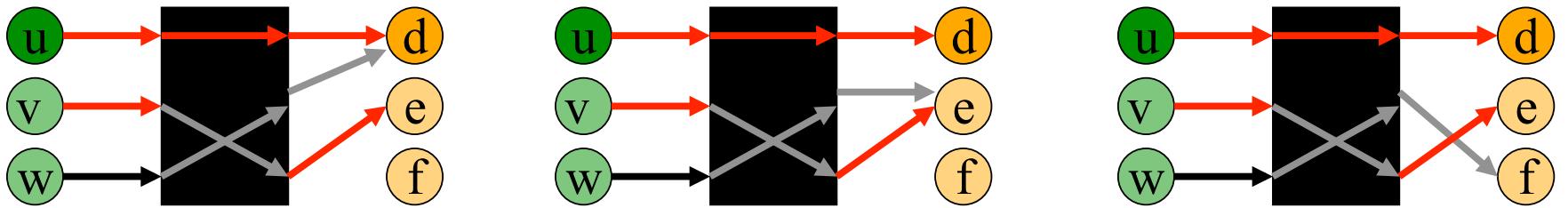
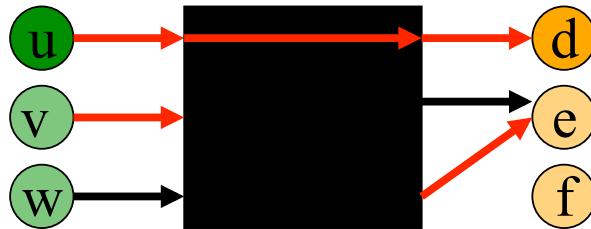
- Each user v selects a destination from distribution p^v

Probabilistic Black-box



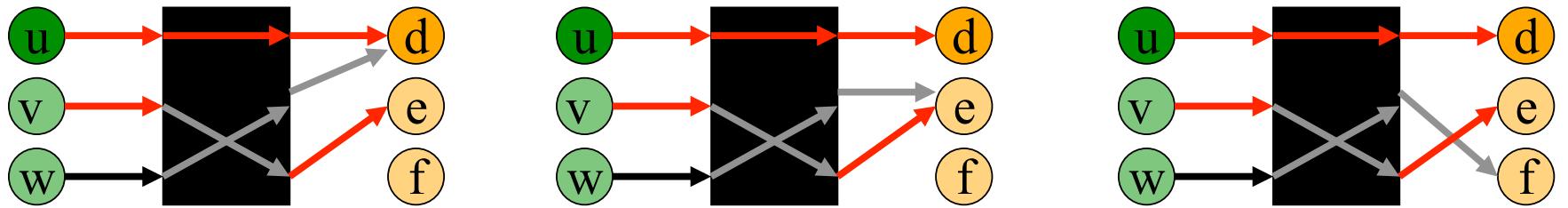
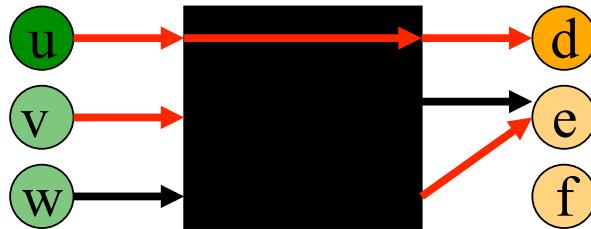
- Each user v selects a destination from distribution p^v
- Inputs and outputs are observed independently with probability b

Probabilistic Anonymity



Indistinguishable
configurations

Probabilistic Anonymity



Indistinguishable
configurations

Conditional distribution: $\Pr[u \rightarrow d] = 1$

Black Box Model

Let U be the set of users.

Let Δ be the set of destinations.

Configuration C

- User destinations $C_D : U \rightarrow \Delta$
- Observed inputs $C_I : U \rightarrow \{0,1\}$
- Observed outputs $C_O : U \rightarrow \{0,1\}$

Let X be a random configuration such that:

$$\Pr[X=C] = \prod_u p_{C_D(u)}^u \cdot b^{C_I(u)} (1-b)^{1-C_I(u)} \cdot b^{C_O(u)} (1-b)^{1-C_O(u)}$$

Probabilistic Anonymity

The metric Y for the unlinkability of u and d in C is:

$$Y(C) = \Pr[X_D(u)=d \mid X \approx C]$$

Probabilistic Anonymity

The metric Y for the unlinkability of u and d in C is:

$$Y(C) = \Pr[X_D(u)=d \mid X \approx C]$$

Note: There are several other candidates for a probabilistic anonymity metric, e.g. **entropy**

Probabilistic Anonymity

The metric Y for the unlinkability of u and d in C is:

$$Y(C) = \Pr[X_D(u)=d \mid X \approx C]$$

Exact Bayesian inference

- Adversary after long-term intersection attack
- Worst-case adversary

Probabilistic Anonymity

The metric Y for the unlinkability of u and d in C is:

$$Y(C) = \Pr[X_D(u)=d \mid X \approx C]$$

Exact Bayesian inference

- Adversary after long-term intersection attack
- Worst-case adversary

Unlinkability given that u visits d :

$$\mathbf{E}[Y \mid X_D(u)=d]$$

Worst-case Anonymity

Worst-case Anonymity

Let $p^u_1 \geq p^u_2 \geq p^u_{d-1} \geq p^u_{d+1} \geq \dots \geq p^u_\delta$

Theorem 1: The maximum of $\mathbf{E}[Y | X_D(u)=d]$ over $(p^v)_{v \neq u}$ occurs when

1. $p^v_\delta = 1$ for all $v \neq u$ OR
2. $p^v_d = 1$ for all $v \neq u$

Worst-case Estimates

Let n be the number of users.

Worst-case Estimates

Let n be the number of users.

Theorem 2: When $p^v_\delta = 1$ for all $v \neq u$:

$$\begin{aligned} E[Y | X_D(u)=d] &= b + b(1-b)p^u_d + \\ &(1-b)^2 p^u_d \left[(1-b)/(1-(1-p^u_\delta)b) \right] + O(\sqrt{\log n / n}) \end{aligned}$$

Worst-case Estimates

Let n be the number of users.

Theorem 2: When $p^v_\delta = 1$ for all $v \neq u$:

$$\begin{aligned} \mathbb{E}[Y | X_D(u)=d] &= b + b(1-b)p^u_d + \\ &(1-b)^2 p^u_d [(1-b)/(1-(1-p^u_\delta)b)) + O(\sqrt{\log n}/n)] \\ &\approx b + (1-b) p^u_d \end{aligned}$$

Worst-case Estimates

Let n be the number of users.

Theorem 2: When $p^v_\delta = 1$ for all $v \neq u$:

$$\begin{aligned} \mathbb{E}[Y | X_D(u)=d] &= b + b(1-b)p^u_d + \\ &(1-b)^2 p^u_d [(1-b)/(1-(1-p^u_\delta)b)) + O(\sqrt{\log n}/n)] \\ &\approx b + (1-b) p^u_d \end{aligned}$$

$$\mathbb{E}[Y | X_D(u)=d] \geq b^2 + (1-b^2) p^u_d$$

Worst-case Estimates

Let n be the number of users.

Theorem 2: When $p^v_\delta = 1$ for all $v \neq u$:

$$\begin{aligned} E[Y | X_D(u)=d] &= b + b(1-b)p^u_d + \\ &(1-b)^2 p^u_d [(1-b)/(1-(1-p^u_\delta)b)) + O(\sqrt{\log n}/n)] \\ &\approx b + (1-b) p^u_d \end{aligned}$$

$$E[Y | X_D(u)=d] \geq b^2 + (1-b^2) p^u_d$$

Increased chance of total compromise from b^2 to b .

Worst-case Estimates

Let n be the number of users.

Theorem 2: When $p^v_d = 1$ for all $v \neq u$:

$$\begin{aligned} \mathbb{E}[Y | X_D(u)=d] &= b + b(1-b)p^u_d + \\ &(1-b)^2 p^u_d [(1-b)/(1-(1-p^u_d)b)) + O(\sqrt{\log n}/n)] \end{aligned}$$

Theorem 3: When $p^v_d = 1$ for all $v \neq u$:

$$\begin{aligned} \mathbb{E}[Y | X_D(u)=d] &= b^2 + b(1-b)p^u_d + \\ &(1-b)p^u_d/(1-(1-p^u_d)b) + O(\sqrt{\log n}/n) \end{aligned}$$

Typical Case

Let each user select from the Zipfian distribution:
 $p_{d_i} = 1/(\mu i^s)$ (Has been shown web destinations follow Zipf distribution.)

Theorem 4:

$$\mathbb{E}[Y | X_D(u)=d] = b^2 + (1 - b^2)p_d^u + O(1/n)$$

Typical Case

Let each user select from the Zipfian distribution:

$$p_{d_i} = 1/(\mu i^s)$$

Theorem 4:

$$\mathbf{E}[Y | X_D(u)=d] = b^2 + (1 - b^2)p^u_d + O(1/n)$$

Theorem proof does not depend on particular distribution as much as that it is the same distribution across users.

Summary of probabilistic analysis

1. Used a black-box abstraction to create a probabilistic model of onion routing
2. Analyzed unlinkability
 - a. Provided worst-case bounds
 - b. Examined a typical case

Potential Future Work

1. Extend analysis to other types of anonymity and to other systems.
2. Examine how quickly users distribution are learned.
3. Analyze entry guard choice.
 - If sensitive destinations are rare, maybe better not using guards?

What is a Hidden Server?

- Alice can connect to Bob's server without knowing where it is or possibly who he is
- Who needs this?



UNDERMINING FREEDOM OF EXPRESSION IN CHINA

The role of Yahoo!, Microsoft and Google



Amnesty International

MSN Spaces in China

- MSN blocked search results and creation of blog titles with “democracy”, “human rights”, and “freedom of expression”.
- Dec. 2005: MSN Spaces yanked the blog of Zhao Jing (Michael Anti) both in China and globally
- Later changed policy to only remove access from China and only after formal legal notice

What's being done against censorship?

The screenshot shows the homepage of Irrepressible.info, a campaign by Amnesty International. The page features a green header bar with the site's name and a megaphone icon. Below the header, there are navigation links for Welcome, About, and Participate. A prominent green box on the right displays the number 65536, indicating the count of people who have signed the pledge. The main content area discusses internet censorship, mentioning specific cases like the arrest of Salih in Uzbekistan. The footer contains a call to action about undermining censorship through publishing fragments.

start Central Authentication Service PET 2007 We... Main page GPL-like patent license Livres et métiers ABSOLUTE Mac Noble Duty ...Other

welcome | irrepressible.i... × welcome | irrepressible.i...

English Español Français عربي

IRREPRESSIBLE.INFO
an Amnesty International campaign

Welcome About Participate

Irrepressible
Adj. 1) Impossible to repress or control.

Chat rooms monitored. Blogs deleted. Websites blocked. Search engines restricted. People imprisoned for simply posting and sharing information.

The Internet is a new frontier in the struggle for human rights. Governments – with the help of some of the biggest IT companies in the world – are cracking down on freedom of expression.

Amnesty International, with the support of The Observer UK newspaper, is launching a campaign to show that online or offline the human voice and human rights are impossible to repress.

65536
people have signed the pledge

“... She said it was absurd that Uzbek President Islam Karimov was free to travel to Europe while Salih remained subject to an Interpol arrest warrant. ...”

This is an excerpt from:
<http://www.erkinyurt.org/>

The site belongs to United Uzbek Democratic Coalition , and has been censored in Uzbekistan. It is a pro-democracy website.

Undermine censorship by publishing irrepressible fragments

It's not just about access to information

**REPORTERS
WITHOUT BORDERS**
FOR PRESS FREEDOM

Français Espanol عربي

News from around the world

Africa | Americas | **Asia** | Europe and the former USSR

Middle East and North Africa | Internet | United Nations

RSS Search OK DONATION

Work with us About us Special reports Regular reports Media downloads



CHINA 19 April 2006

Asia press releases

Yahoo ! implicated in third cyberdissident trial

US company's collaboration with Chinese courts highlighted in Jiang Lijun case

Reporters Without Borders has obtained a copy of the verdict in the case of Jiang Lijun, sentenced to four years in prison in November 2003 for his online pro-democracy articles, showing that Yahoo! helped Chinese police to identify him.

It is the third such case, following those of Shi Tao and Li Zhi, proving the implication of the American Internet company.

The verdict, made available and translated into English by the human rights group, the Dui Hua Foundation, can be downloaded below.

"Little by little we are piecing together the evidence for what we have long suspected, that Yahoo! is implicated in the arrest of most of the people that we have been defending," the press freedom organisation said.

"Last week we went to the headquarters of the company to urge

IN THIS COUNTRY

- 20 April - China Member of uighur minority sentenced to nine years in jail for trying to post "secessionist" articles
- 30 March - China French website blocked for warning of risks of investing in China
- 22 March - China Disturbing lapses in application of new rules for foreign media
- 19 March - China Cyber-dissident Zhang Jianhong ("Li Hong") gets six years in prison
- 23 February - China One year taken off former newspaper president's prison sentence

IN THE ANNUAL REPORT

- China - Annual report 2006
- China - Annual report 2005
- China - 2004 Annual Report
- China - 2003 Annual report

[Tor Test Results](#)[Absolute Mac](#)[Apple](#)[News \(721\) ▾](#)[Apple \(137\) ▾](#)[Image:ChinaGeography...](#)[A Silk Road Timeline](#)[Yahoo! link confirmed in...](#)[6th Worl...](#)

Search:

Wired News

Search

[Top](#)[Technology](#)[Culture](#)[Politics](#)[News Wires](#)[Blogs](#)[Columns](#)[Wired Mag](#)

Iran Cracks Down on Bloggers

[PRINT](#)[MAIL](#)[RANTS + RAUES](#)

Associated Press 16:13 PM Mar, 28, 2006

DUBAI, United Arab Emirates -- On his last visit to Iran, Canadian-based blogger Hossein Derakhshan was detained and interrogated, then forced to sign a letter of apology for his blog writings before being allowed to leave the country. Compared to others, Derakhshan is lucky.

Dozens of Iranian bloggers have faced harassment by the government, been arrested for voicing opposing views, and fled the country in fear of prosecution over the past

Breaking

Breaking News from AP and Re

- Charges Filed in NYC Fire; 2 ir...
- Md. Judge to Let Muhammad...
- World powers to discuss next...
- American Airlines Sued Over I...
- Statehouses Take Up Immigr...

It's not only
about
dissidents in
faraway
lands

do delawareonline

News ▶ Business ▶ Sports ▶ Opinion ▶ Entertainment ▶ Life ▶ Video ▶

SEARCH/Delaware ▶ All ▶

Advertisement

HOME > BUSINESS

Freedom of speech? ... better ask your boss

The First Amendment takes on a different role when applied to the workplace

By GARY HABER, *The News Journal*

Convinced you have freedom of speech at work? Think again.

Maybe you should ask the AstraZeneca pharmaceutical sales manager fired earlier this month for comments he reportedly made in a company newsletter comparing physicians' offices to "a big bucket of money."

Or, the Utah Web designer fired for observations about her job she posted on her personal blog.

Or, former Philadelphia Eagles wide receiver Terrell Owens, whose pointed criticism of the team and its quarterback got him suspended in 2005.

The First Amendment experts are quick to point out, doesn't

Subscribe
Email Story
Print Story
Discuss Story

Top Story Chat

- Jury finds in favor of officers in wrongful death case - 64 Comments

News Choices

- Get Published
- Webcasts
- Wireless
- Text Alerts
- RSS Feeds
- News Archive


The News Journal/HOWARD JOHNSON

EFF Blogging Tips

(from *Delaware Online April 23, '07*)

TIPS FOR BLOGGING ABOUT JOB

The Electronic Frontier Foundation, a group that protects the rights of bloggers and other Internet users, offers some tips for blogging about work:

- Don't blog using office computers.
- Use a pseudonym for yourself, and don't identify your employer by name.
- Don't include details about the company from which a reader can figure out who you work for.
- Don't post pictures of yourself on your blog, by which someone can figure out who you are.
- Consider using a service like invisiblog.com, which hosts anonymous blogs for free, or [LiveJournal](http://LiveJournal.com), which restricts access to your blog to those with a password or to people you designate as friends.

Source: *Electronic Frontier Foundation*

invisiblog.net (beta)

anonymous weblog publishing

invisiblog.net lets you publish a weblog using [GPG](#) and the [Mixmaster](#) anonymous remailer network. You don't ever have to reveal your identity - not even to us. You don't have to [trust us](#), because we'll never know who you are.

[Learn more, or start your own invisiblog now.](#)

Read system news and updates at [Invisiblog News](#).

Oct 20, 2005: we have moved to a new hosting company, following a request from the old one to censor some content. Web content is back up, but it may be a few days before new posts show up.

new blogs

- Oct 14: [Streikblogblog](#)
- Oct 14: [Questo è il mio blog anonimo](#)
- Oct 09: [Claude Gene Grinder - Life...](#)
- Oct 04: [untitled](#)
- Oct 03: [My anonymous blog](#)
- Sep 30: [Dead Hooker Project](#)
- Sep 30: [Inside The Bank](#)
- Sep 28: [Thought you should know...](#)
- Sep 28: [The New Hypocrisy](#)
- Sep 27: [antani](#)
- Sep 27: [untitled](#)

updated blogs

- Oct 18: [Retvos](#)
- Oct 15: [Dead Hooker Project](#)
- Oct 15: [untitled](#)
- Oct 11: [nexus](#)
- Oct 10: [Diary of a Paedophile](#)
- Oct 08: [Use this blog to test](#)
- Oct 04: [untitled](#)
- Oct 02: [Inside The Bank](#)
- Sep 29: [antani](#)
- Sep 29: [The New Hypocrisy](#)
- Sep 27: [Melbourne IT Mela](#)

NearlyFreeSpeech.NET – Unknown Site – invisiblog.com

OneStart Central Authentication Service PET 2007 We... Main page GPL-like patent license Livres et métiers Absolute Mac

North Carolina... Onion Routing delawareonlin... NearlyFreeSpe... Dummies:Taki... Jiang Lijun: Inf... BBC NEWS | Te...

NearlyFreeSpeech.NET

Unknown Site: invisiblog.com

You have requested a server name that our network does not recognize. There are several possible reasons why this might happen:

- The site does not exist, or if it did exist, it has been permanently deleted by the site owner.
- The site owner may have configured a third party DNS service to point this name at our network without telling our network what web site this name is associated with.
- A problem with our network prevented us from recognizing this site.
- The application you used to retrieve this page is not compliant with the HTTP standard.

If you are one of our members and you expected to see your page here, please visit our [Member Home Page](#) and check the following:

- Make sure you have added the name **invisiblog.com** to your site as an alias.
- Make sure your site is enabled.
- Try the [Site Troubleshooting Wizard](#).
- If your account recently ran out of funds, check [this FAQ entry](#).

Limits of irrepressible.info and invisiblog.com

- invisiblog must be hosted somewhere that is not
 - censored or blocked or abandoned
- Same for site of censored information
irrepressible.info points at
 - censored websites about Uzbekistan can be pointed at by irrepressible.info but not from Uzbekistan or seen from Uzbekistan
 - site must be anonymized to keep originators
 - Out of prison
 - Employed

Popular Pages

[New Podcasts](#)
[Top 25 Podcasts](#)
[Top Rated Podcasts](#)
[Search for a Podcast](#)
[100 Most Recent Podcasts](#)
[Podcast News Archive](#)
[Podcasting Jobs](#)

Check out Some Podcasts!

[Beyond Jazz](#)
[Bicyclemark's Audiocommunique](#)
[CatFish Show](#)
[Chub Creek](#)
[Concert Blast!](#)
[Disposable Radio](#)
[Domestic Life](#)
[The DV Show](#)
[EarthCore](#)
[EZHelp](#)
[Florn.net](#)
[Gardner Writes](#)
[Hoboken Rock City](#)
[Incoherent Mumblings](#)
[Indie Airplay](#)
[Israelisms](#)
[Kickbiking Podcasts](#)
[Media Artist Secrets](#)

« [Odeo Beta Generates Controversy](#) | [Main](#) | [Microsoft Censoring Free Speech in China](#) »

Torcasting Lets Podcasters Speak Their Minds Anonymously

June 18, 2005

UndergroundMedia.org has published an [introduction to Torcasting Article](#), a method for publishing content to the web anonymously.

Recent events, including a [podcaster losing his job over his show](#), Microsoft's censoring of messages in China and the revealing of the identity of Deep Throat, highlight the fact that **freedom of speech sometimes requires the freedom to speak**.

UndergroundMedia is an outlet for media information to help empower the average person to take an active role in media.

"In order to have a true voice, one has to also have the ability to be anonymous," notes UndergroundMedia. "Privacy is a key component of freedom of press and expression. This is where Tor comes in."

By combining Torcasting with anonymizing voice effects, podcasters can speak their minds while preserving their anonymity.

About Tor & Torcasting

Tor is an "onion routing" project. An onion router is a way of creating tunnels among a group of servers that encrypt traffic and create multiple layers of security, and in turn privacy, for those connecting to the network.

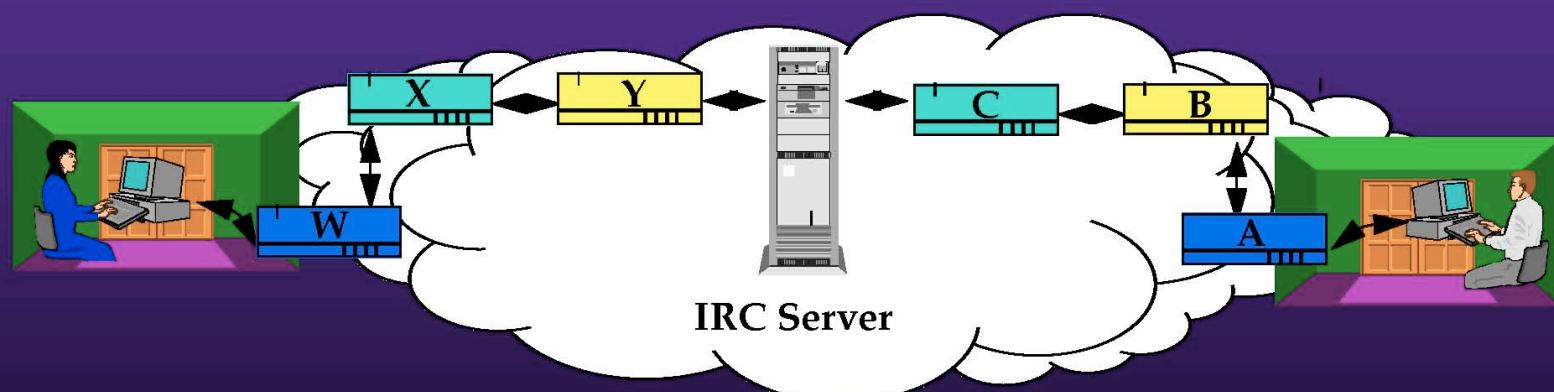
More Hidden Server Applications

- Already extensively discussed
 - Censorship resistant publishers
 - Identity protecting publishing
- Low cost DDoS resistance
- Multilevel secure chat servers
- Automated downgraders of classified docs
- Private location tracking

Other Applications

IRC: Two parties make anonymous connections to an IRC server, which mates the two connections.

Neither party has to trust the other.



Private Location Tracking

Active Badges

Competing Goals:
Track users's location.
But, keep location
information private.

Home station tracks location:

- ◆ Active badge contacts room sensor.
- ◆ Room sensor queries database for a reply onion over an anonymous connection.
- ◆ Sensor contacts home station using reply onion.
- ◆ Home station updates database over an anonymous connection.

Hidden Server Goals

- Servers accessible from anywhere
- Resist attacks from authorized users
- Resist Distributed DoS
- Resist physical attack
- Minimize redundancy, Reduce costs

Location Hidden Servers

- Alice can connect to Bob's server without knowing where it is or possibly who he is
- Already told you why this is desirable, but...
- How is this possible?

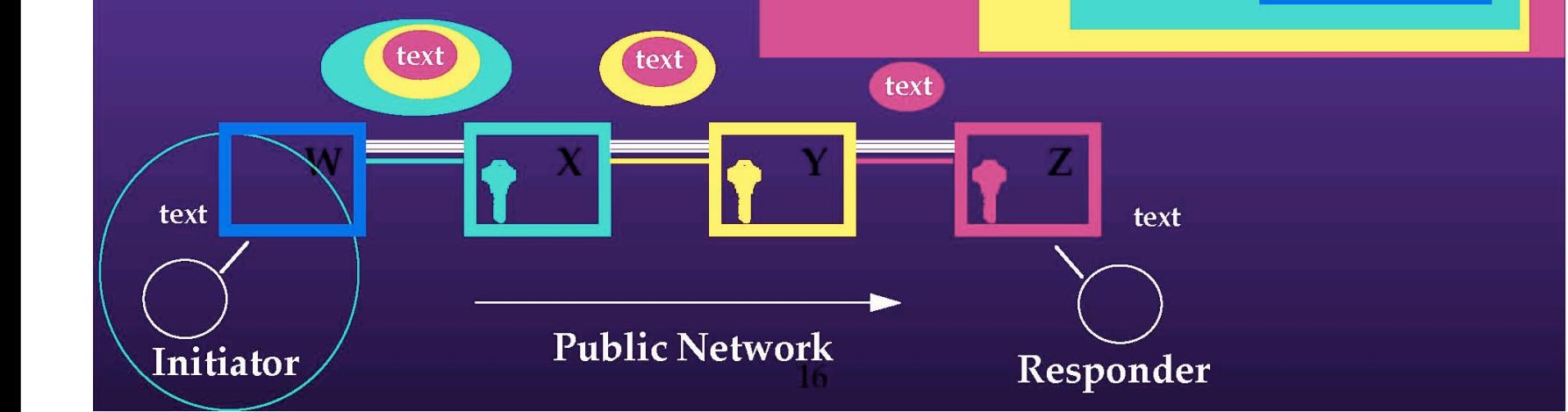
Reply Onions

An Initiator's Onion Routing Proxy can create a Reply Onion that defines a route back to him.

(Z Connect to Y, 

(Y Connect to X, 

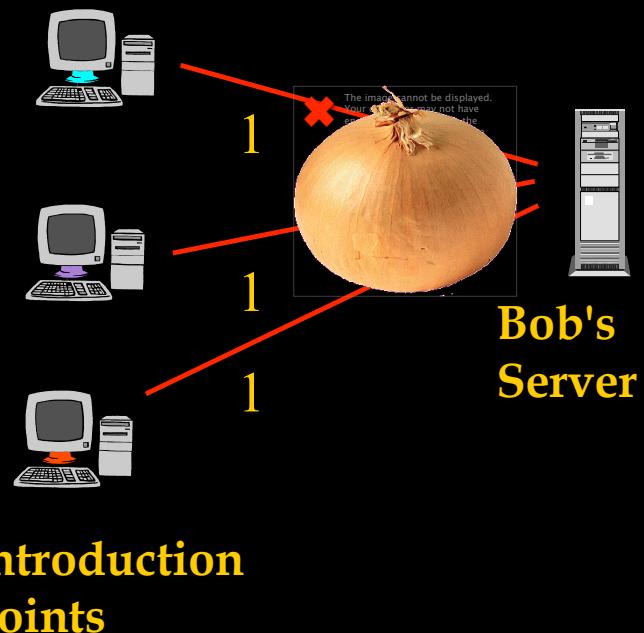
(X Connect to W, 



Location Hidden Servers

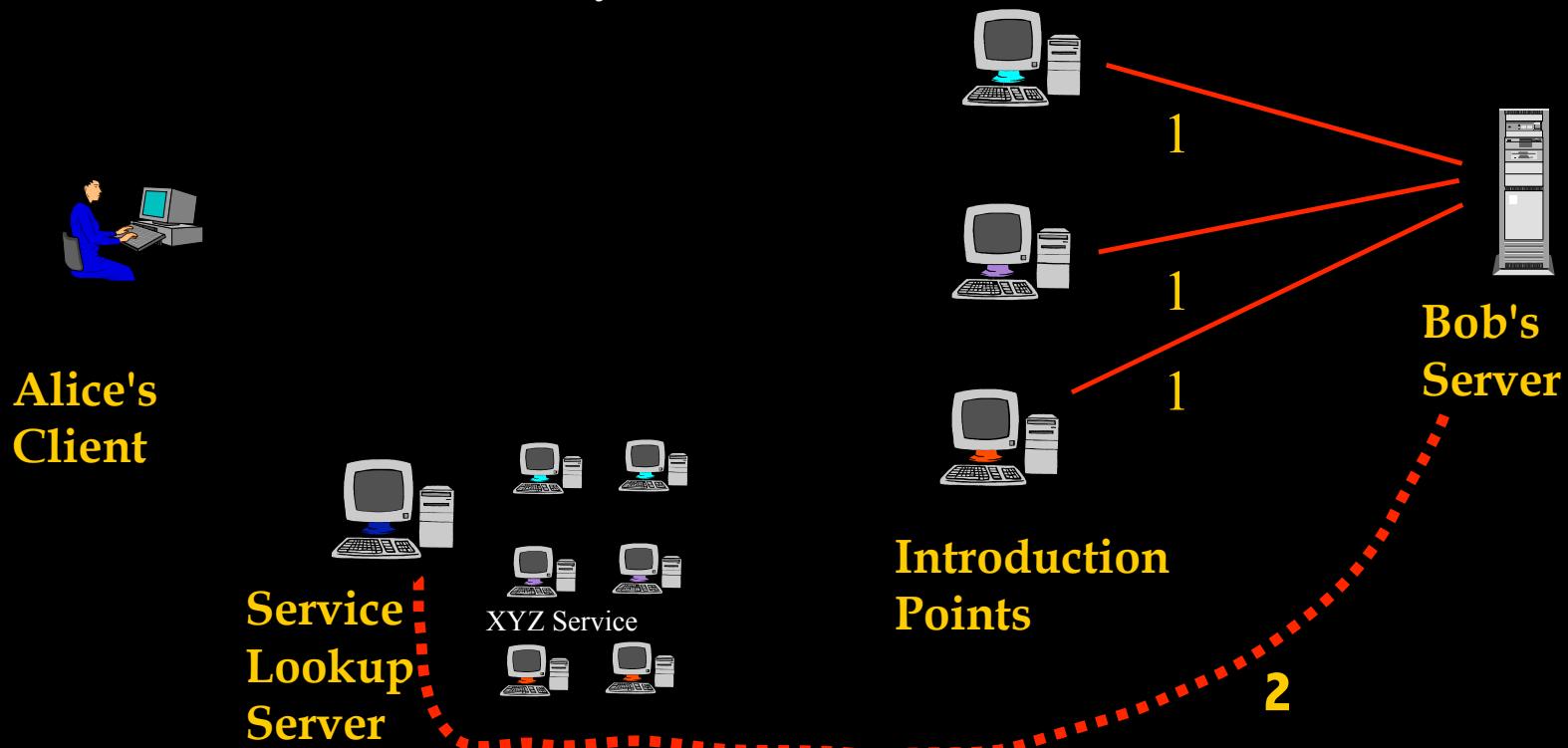
1. Server Bob creates onion routes to [Introduction Points \(IP\)](#)

(All routes in these pictures are onion routed through Tor)



Location Hidden Servers

1. Server Bob creates onion routes to **Introduction Points (IPo)**
2. Bob publishes his xyz.onion address and puts **Service Descriptor** incl. Intro Pt. listed under xyz.onion



FrontPage – Hidden Wiki

http://6sxoyfb3h2nvok2d.onion/tor/

Getting Started Latest Headlines Financial Cryptograph... Measures http://www.ifp.uiuc.e... LEGO Education: Stor... Ask the Wizard! No...

Login Search: Titles Text

Hidden Wiki FrontPage

FrontPage RecentChanges FindPage HelpContents

One if by land, Tor if by wire.

The Hidden Wiki

This [WikiWikiWeb](#) is an example of a location hidden service using [Tor's](#) rendezvous point system.

[Please help save Tor](#). If you have broadband in both directions, it's easy to [run a Tor server](#). If you don't have a fat pipe, encourage others to run their own servers.

Tor Network Status Information: [Xenobite's node list](#) ; [NightEffect Tor Network Status Site](#) ; [Number of Running Tor routers](#)

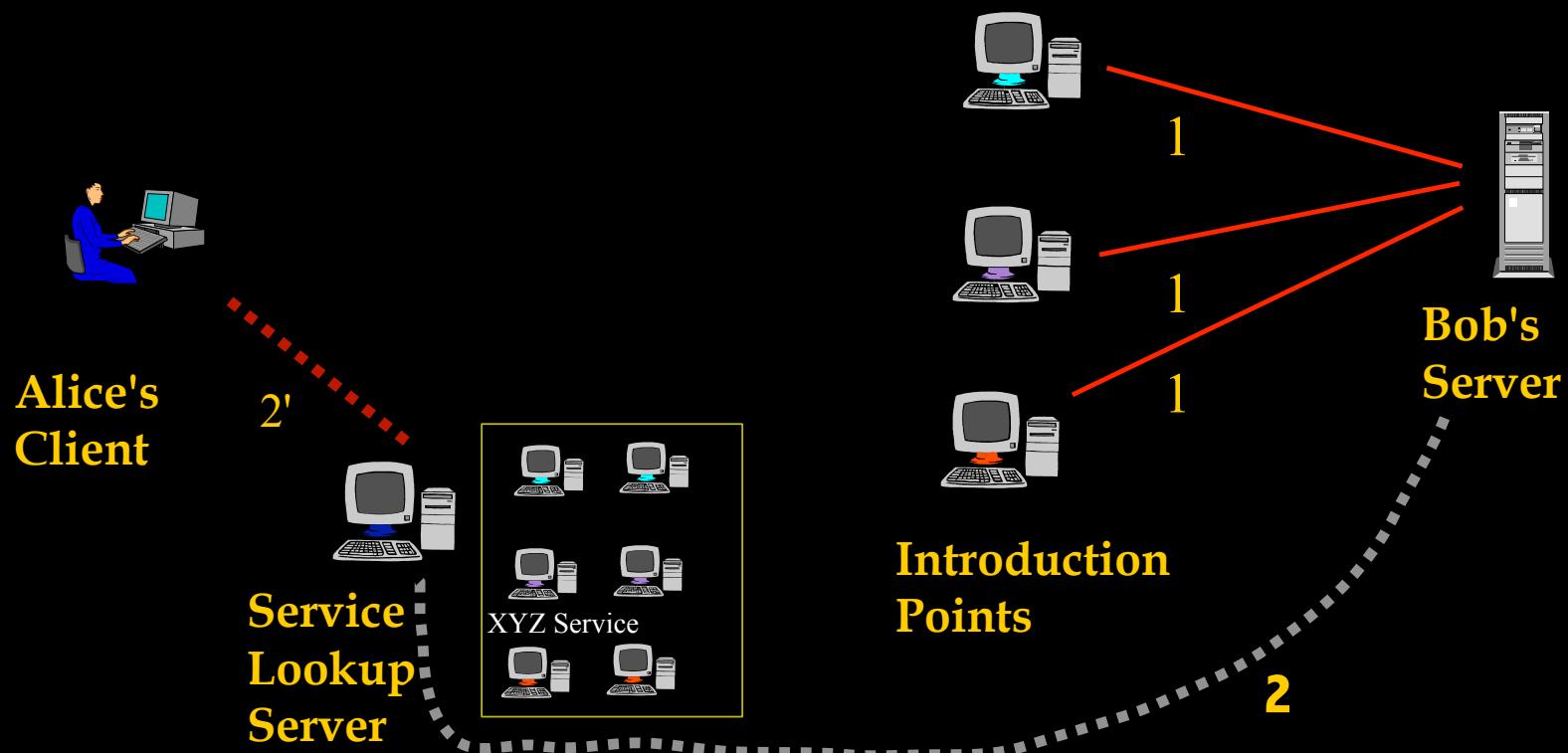
Points of interest include: [TorSetup](#), [QuestionsAndAnswers](#), [WikiMessages](#), [TorizensEmailAddr](#), [FileLinks](#), and [BugMeNot](#).

Useful discussions: [HiddenServiceIdeaPage](#), [LegitimateUses](#), [EmailRecipientAnonymity](#).

- [PreventWorldWarStartCoding](#)
 - [FreeMarketForYou](#)
 - [LongTermGoals](#)
 - [DigitalMoneySupply](#)
 - [AnonymousCurrency](#)
 - [AnonymousVsDebasementControl](#)

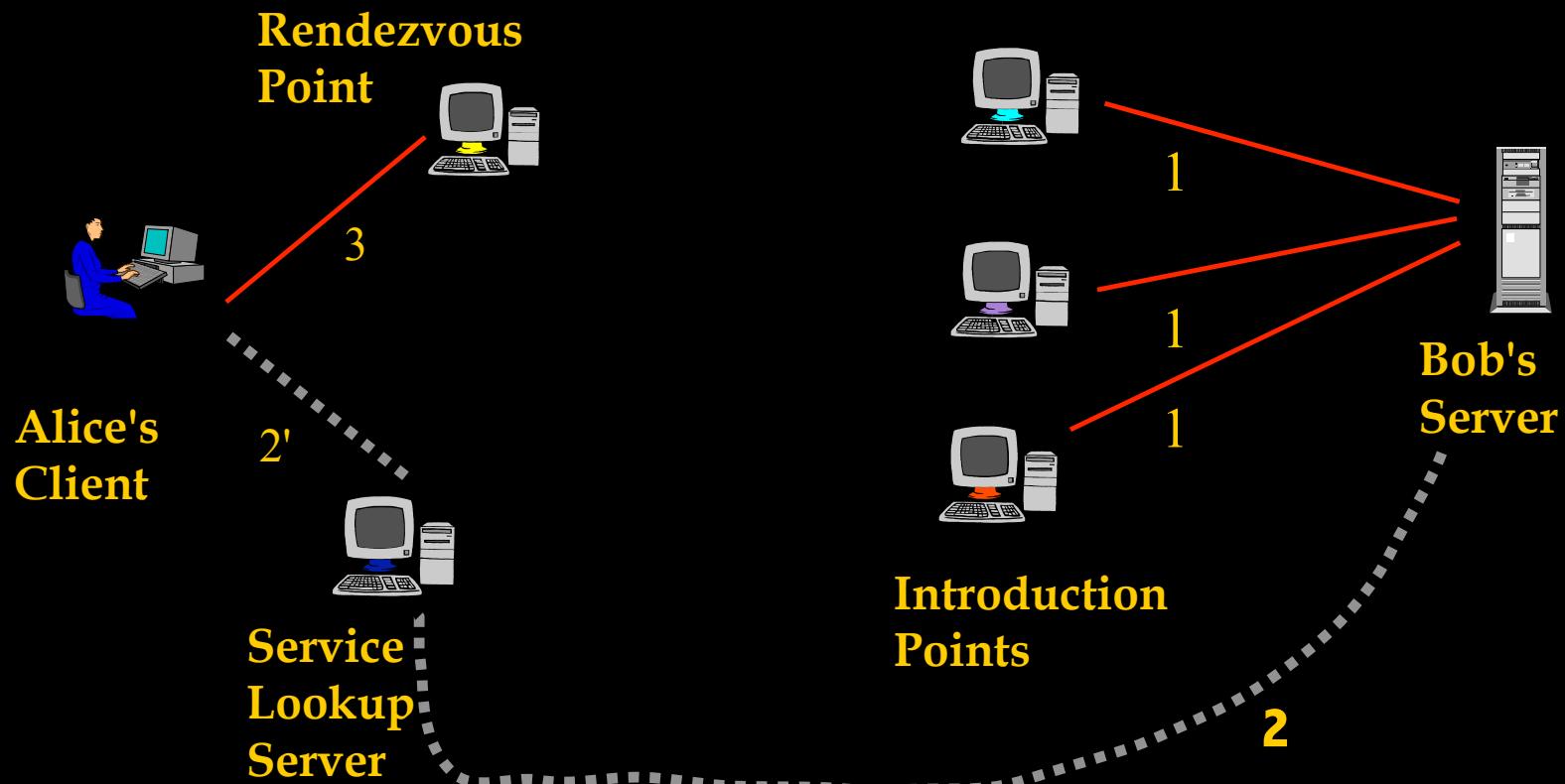
Location Hidden Servers

2'. Alice uses xyz.onion to get Service Descriptor (including Intro Pt. address) at [Lookup Server](#)



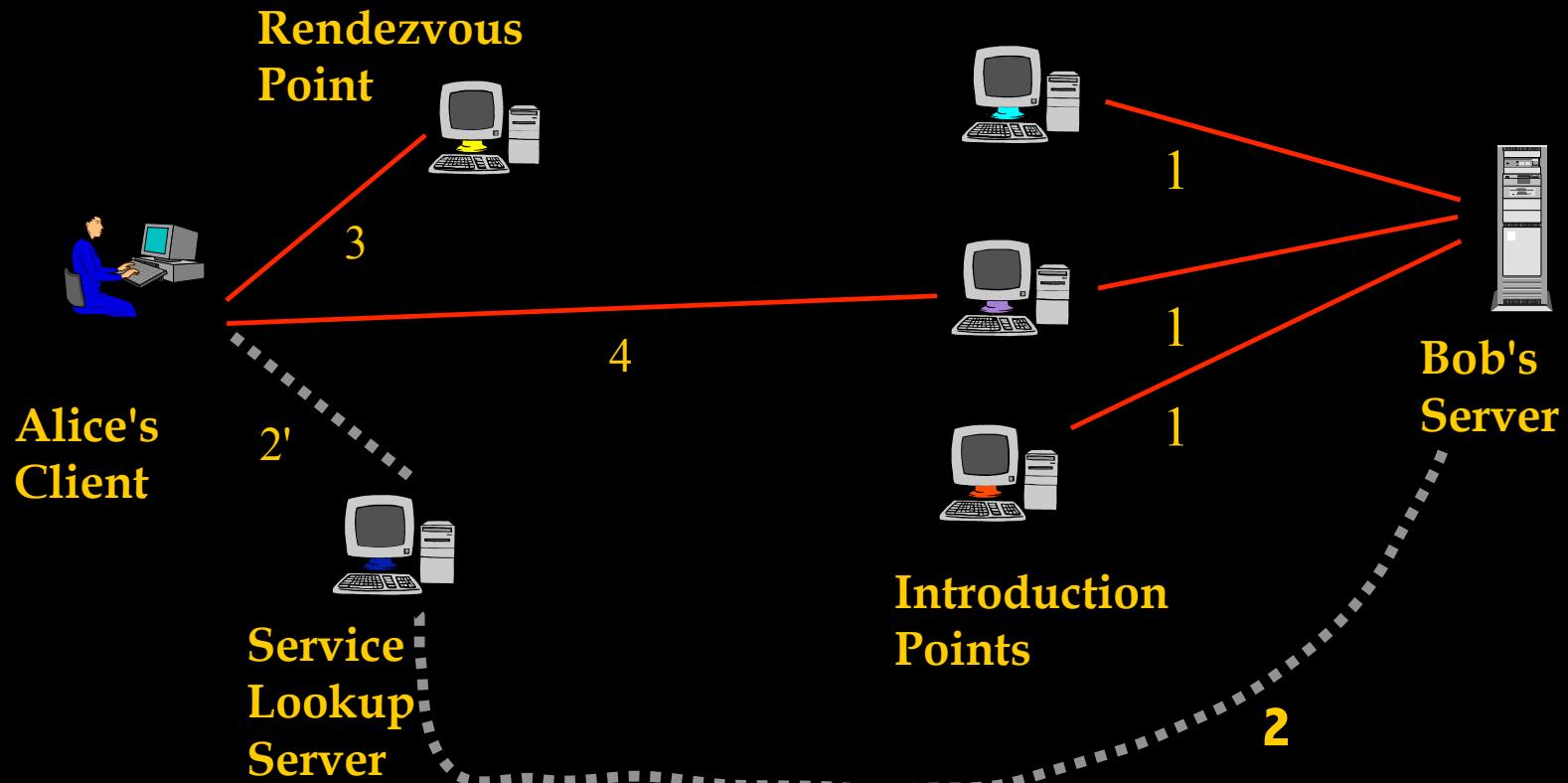
Location Hidden Servers

3. Client Alice creates onion route to Rendezvous Point (RP)



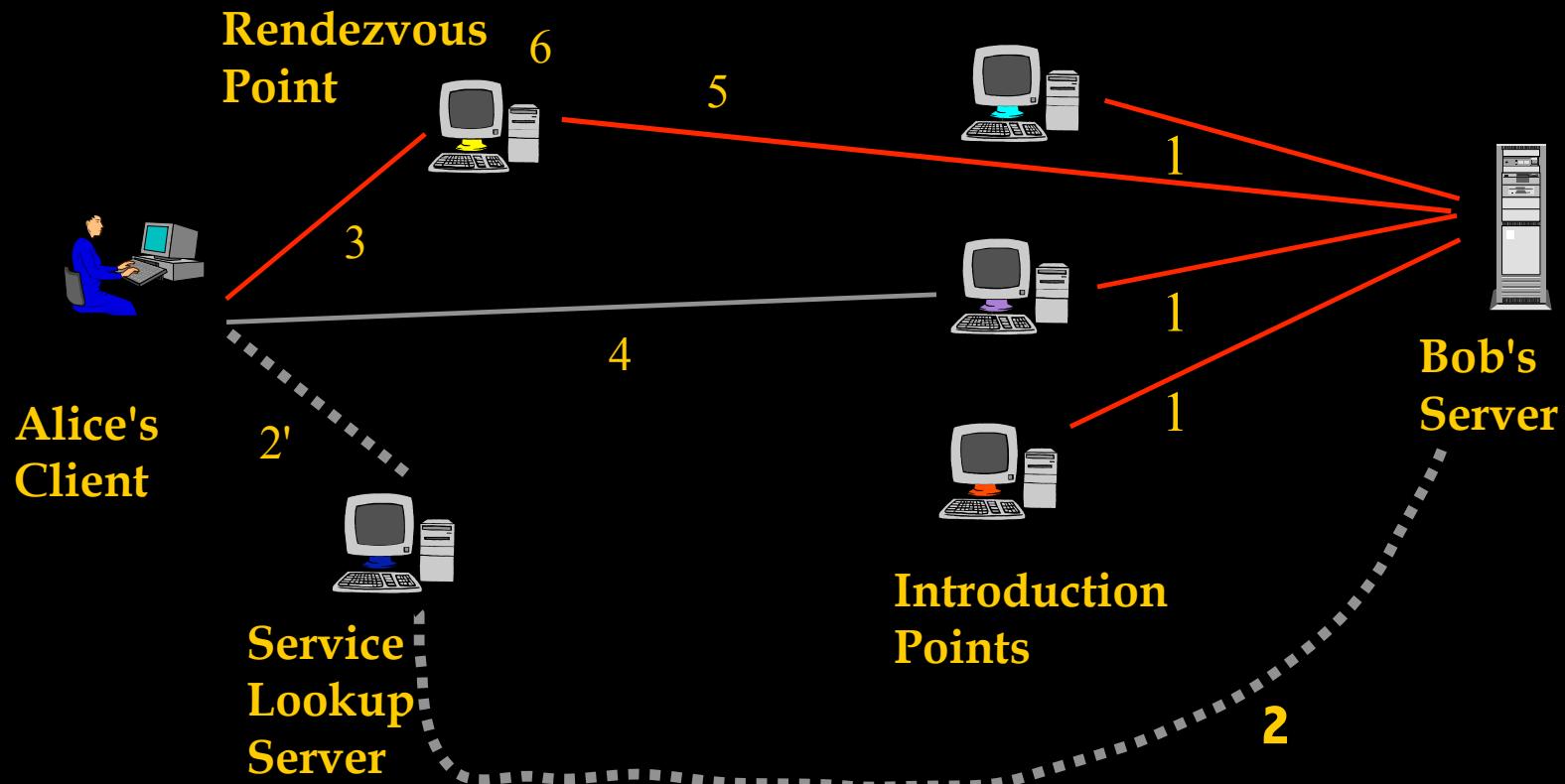
Location Hidden Servers

3. Client Alice creates onion route to Rendezvous Point (RP)
4. Alice sends RP address and any authorization through IPo to Bob



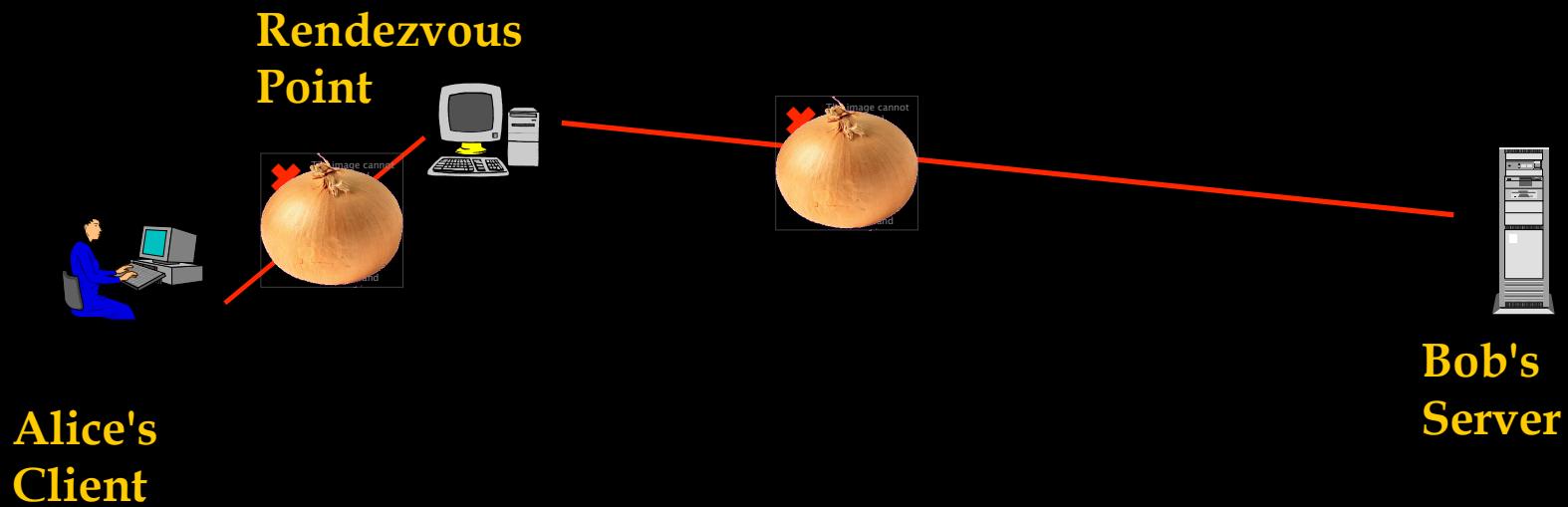
Location Hidden Servers

5. If Bob chooses to talk to Alice, connects to Rendezvous Point
6. Rendezvous Point mates the circuits from Alice and Bob



Location Hidden Servers

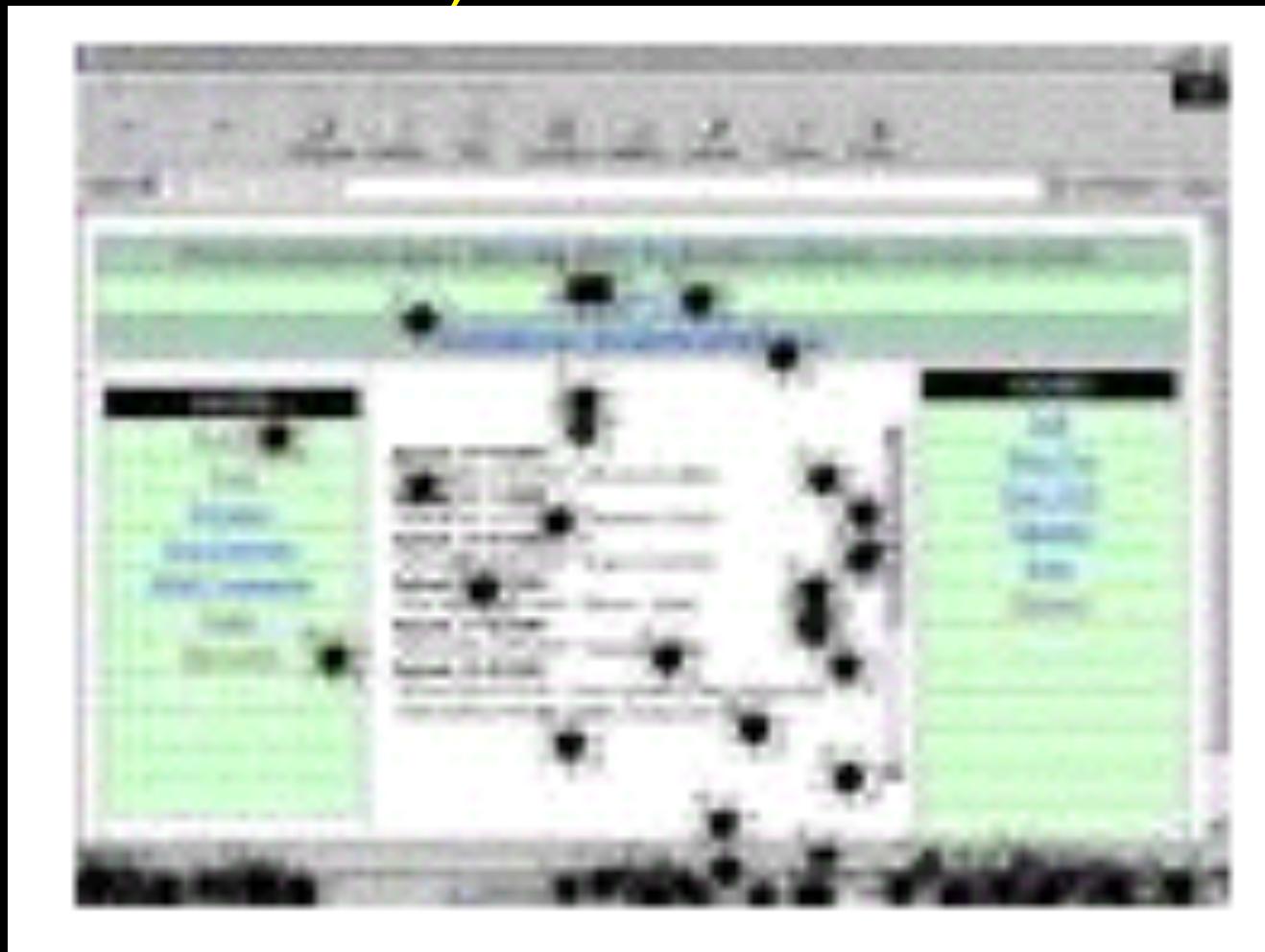
Final resulting communication channel



Attacking Hidden Servers

- In 2006 we showed how to identify a hidden server on the live Tor network in a few minutes to a few hours (depending on configurations) *by owning a single hostile node in the network*
- Note for just the anonymity geeks: This included the first intersection attack of any kind actually conducted on a live network

Attacking Hidden Servers (Not Simulations)



Attacking Hidden Servers (Actual Attacks on Servers in the Wild)

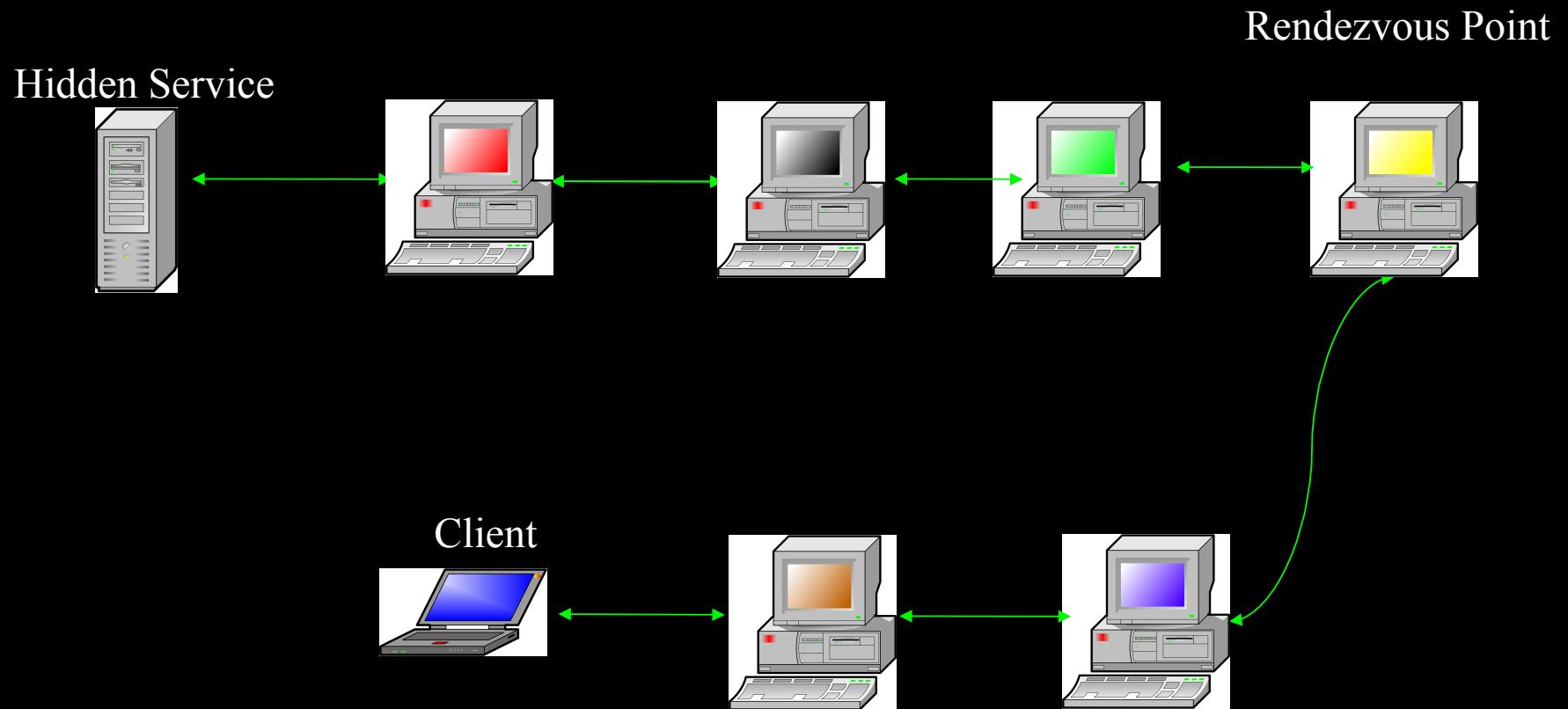


Location Attacks Outline

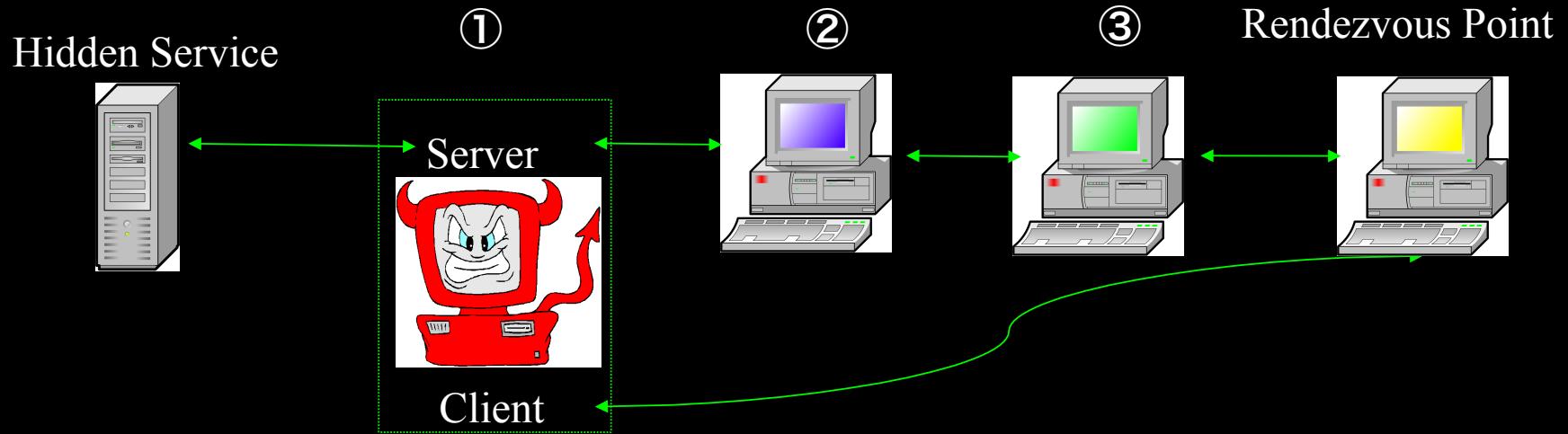
- Attack Overview
 - Phase I: Match the timing signature
 - Phase II: Find node position in circuit
 - Client/Server Separation
 - Intersection
 - Two Node Attack
- Countermeasures (some work -- and some don't)

Normal Scenario Closeup

Tor-connection to Hidden Service



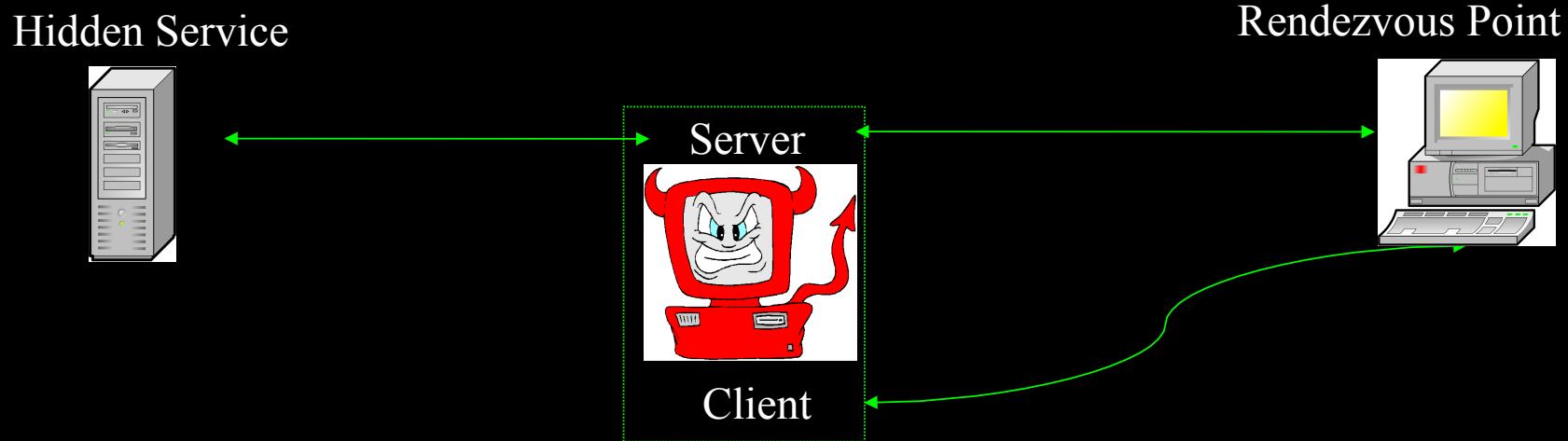
Attack Scenario Closeup



- Using a middle-man Tor server also running a Tor Client
- Make the Client connect directly to Rendezvous Point
- We want to identify the situation shown above
 - Being used as first node by the location hidden service

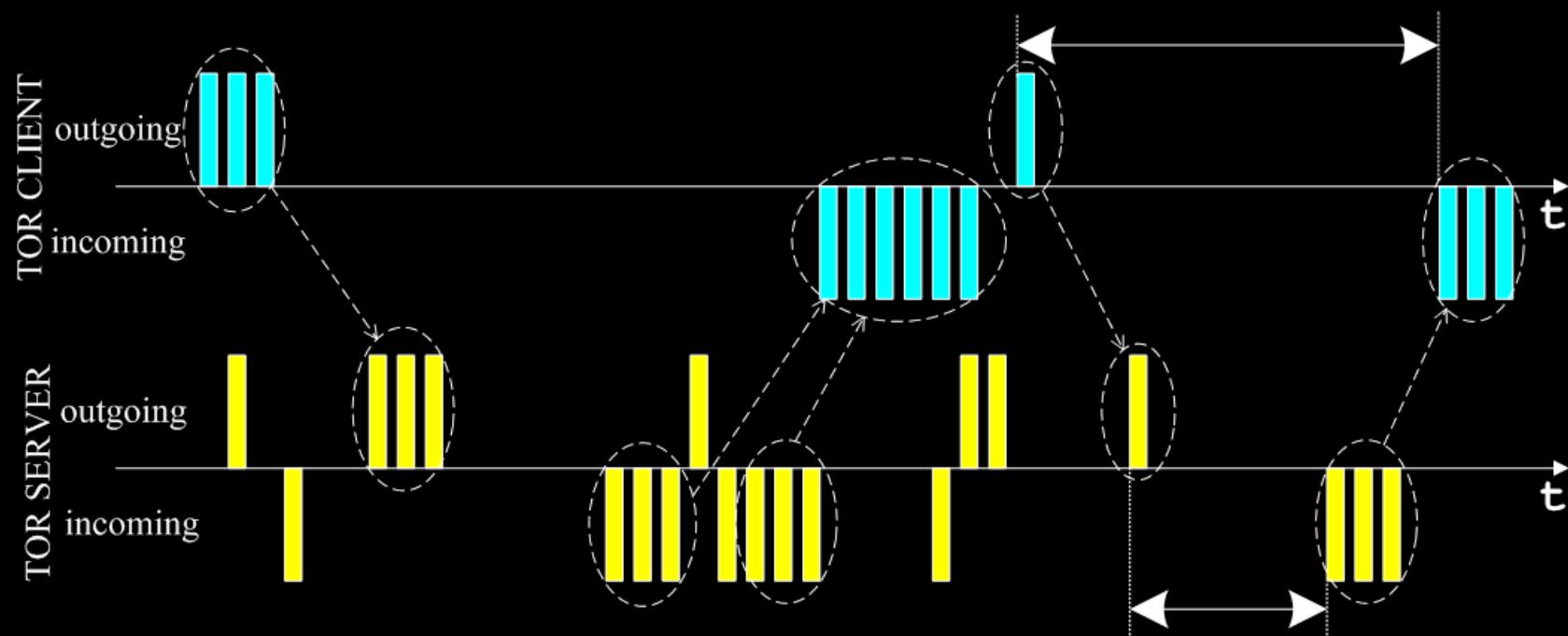
Attack Phase I: Timing

- Client part can create any traffic pattern when sending data
- Response is equally easy to tamper with at server part
- Combination makes circuit “easily” identifiable
- Attacker can know when it is chosen by HS for circuit to RP



Timing Signature

- Client generates data and reads reply from Hidden Server
- Server samples data in all active circuits
- Watch for patterns from Client Part of Node on Server Part



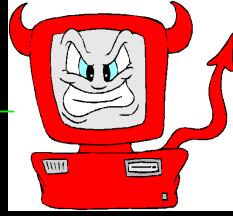
Attack Phase II: Which Position?

Hidden Server

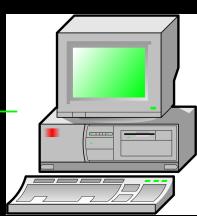
①



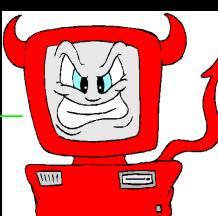
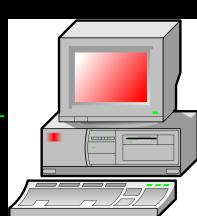
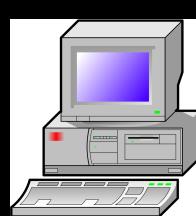
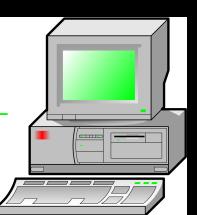
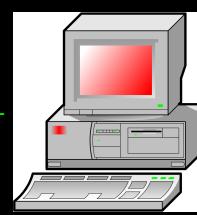
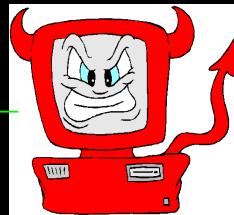
②



③



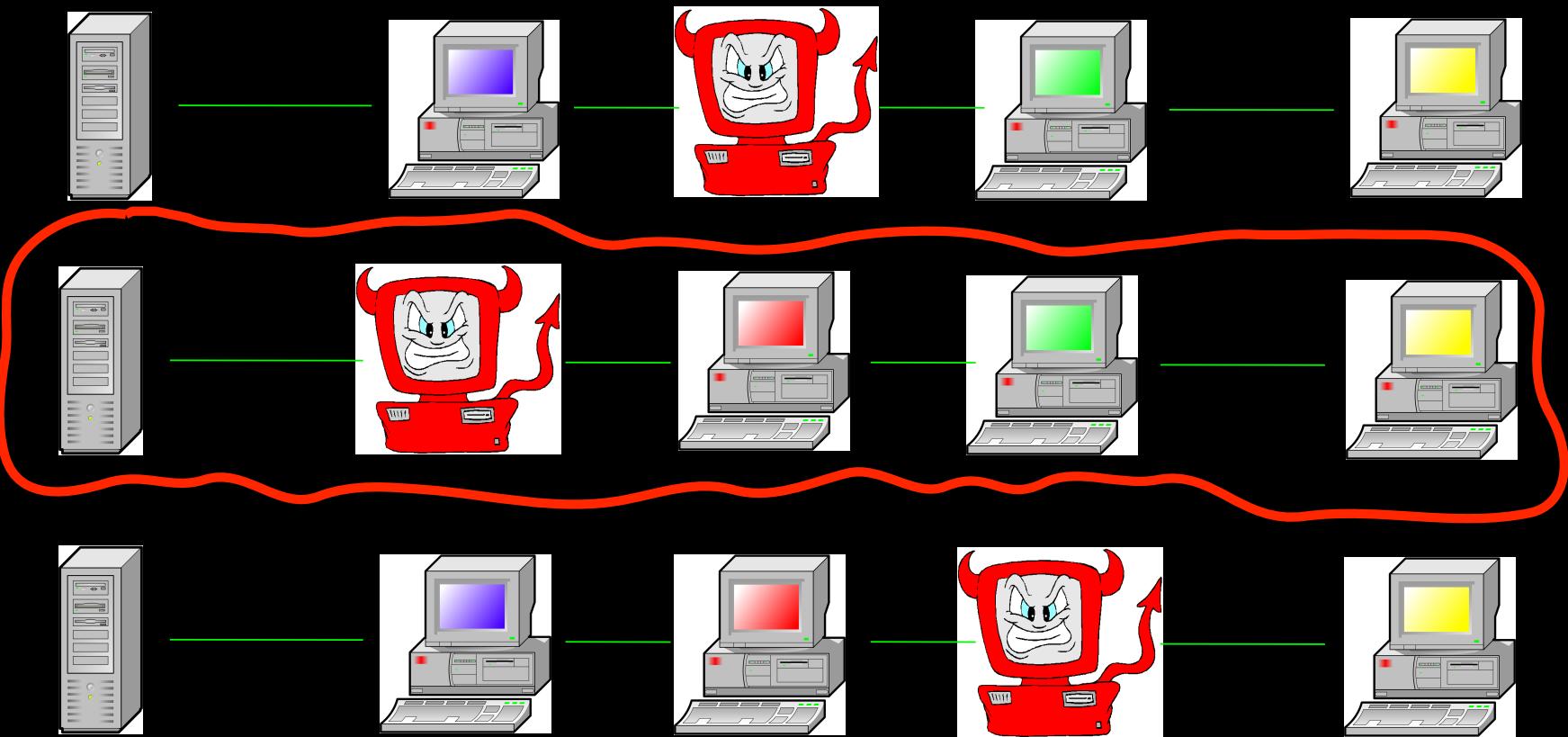
Rendezvous Point



Attack Phase II: Which Position?

Hidden Server

Rendezvous Point



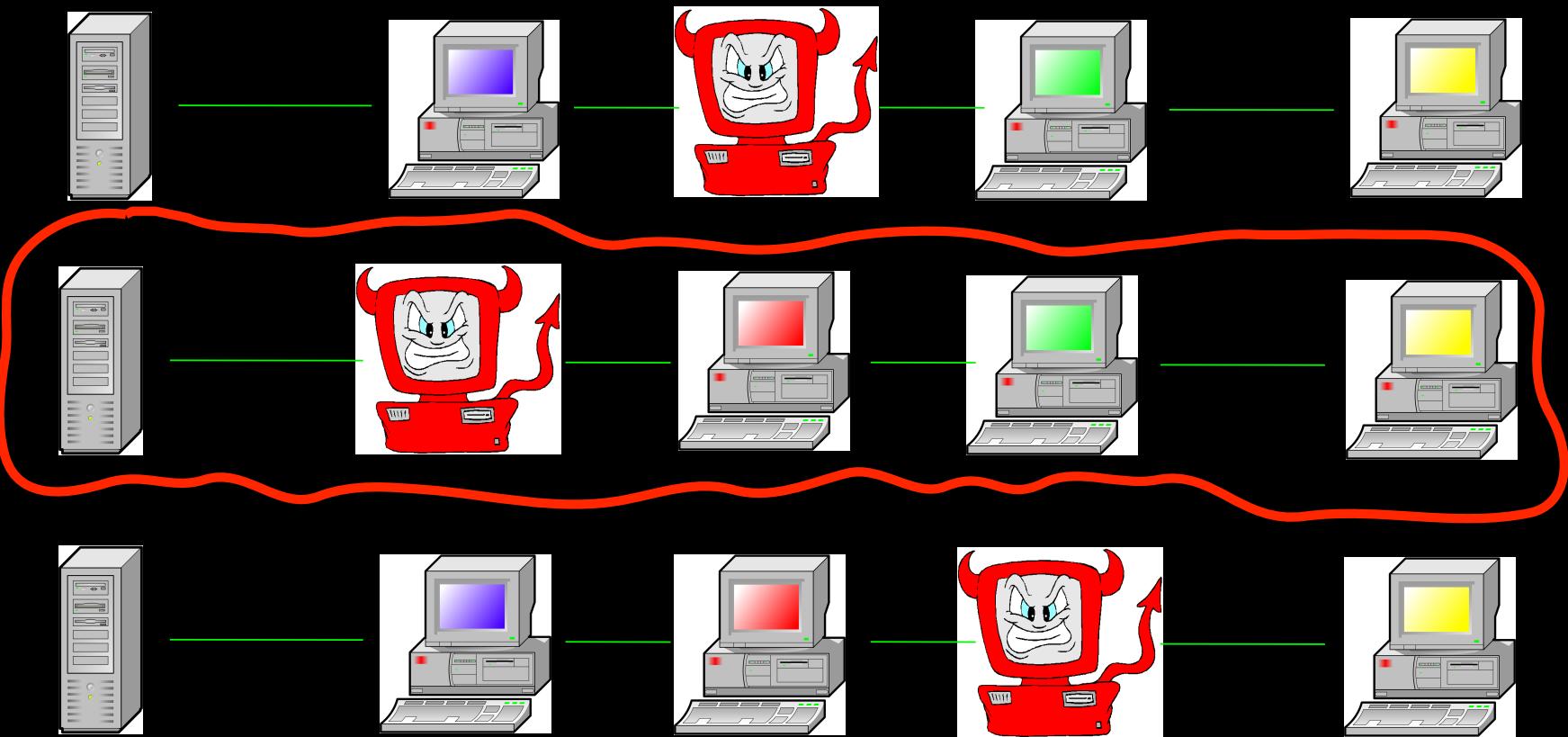
Gotcha!



Attack Phase II: Which Position?

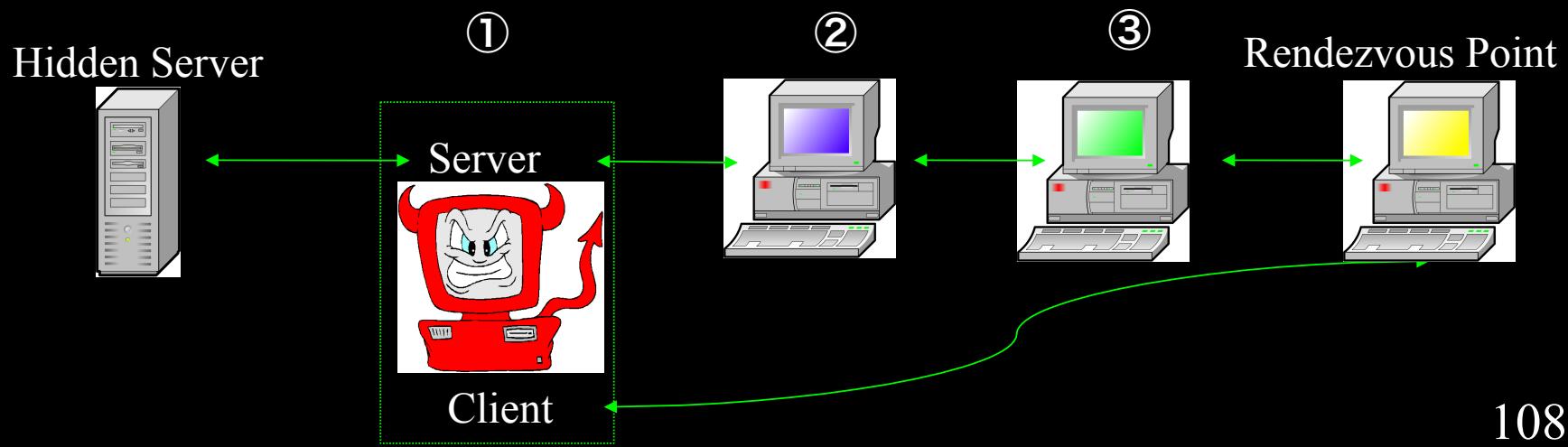
Hidden Server

Rendezvous Point



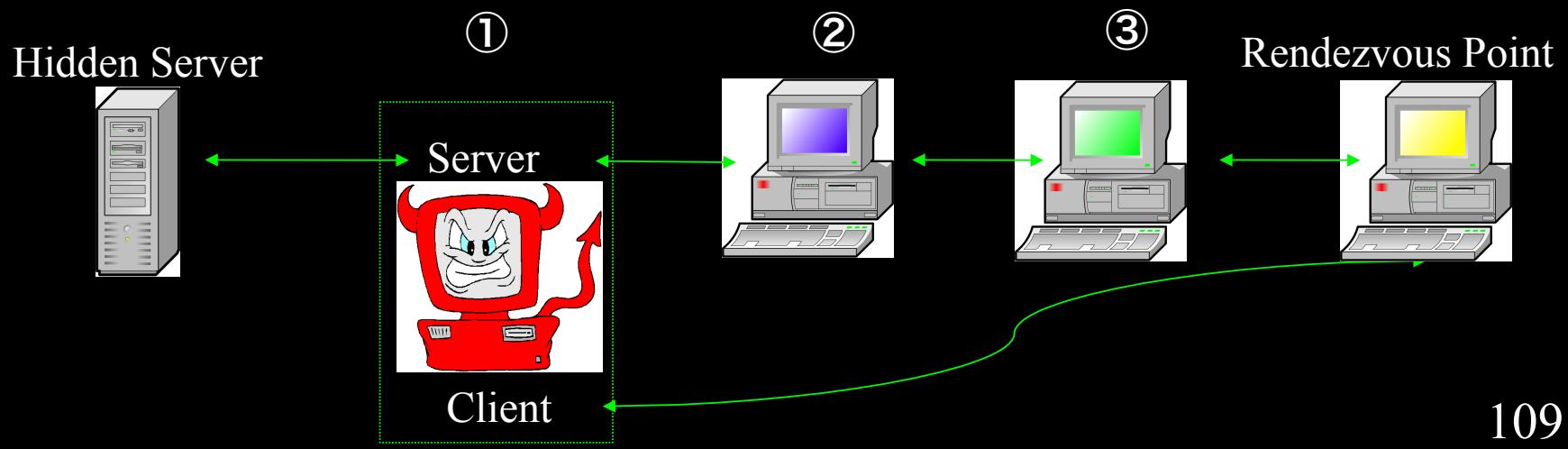
Client/Server Separation

- After confirming participation in circuit by Timing Attack
- Hidden Server as Tor Server
 - Listed. Identifiable as “one of the Tor nodes”
 - Hides hidden service traffic in other Tor traffic
- Hidden Server as external Client
 - Not a part of the listing in the Directory Server
 - Can be used behind a NAT/firewall with ease
- Client is immediately identified if located next to attacker



Intersection Attack

- If identified by Timing Attack as part of a specific circuit:
 - More likely to be contacted by originator than by any other node in circuit
 - One of three positions, 33% chance of either, BUT
 - in first position all connections are from same IP address
 - in second and third position the connections are coming from random nodes
 - Meaning more than 1/3 of all connections are coming from the Hidden Server



Countermeasures

- Dummy Traffic
- Increased path length
- Entry Guard Nodes
 - Random
 - Friend
 - Layered



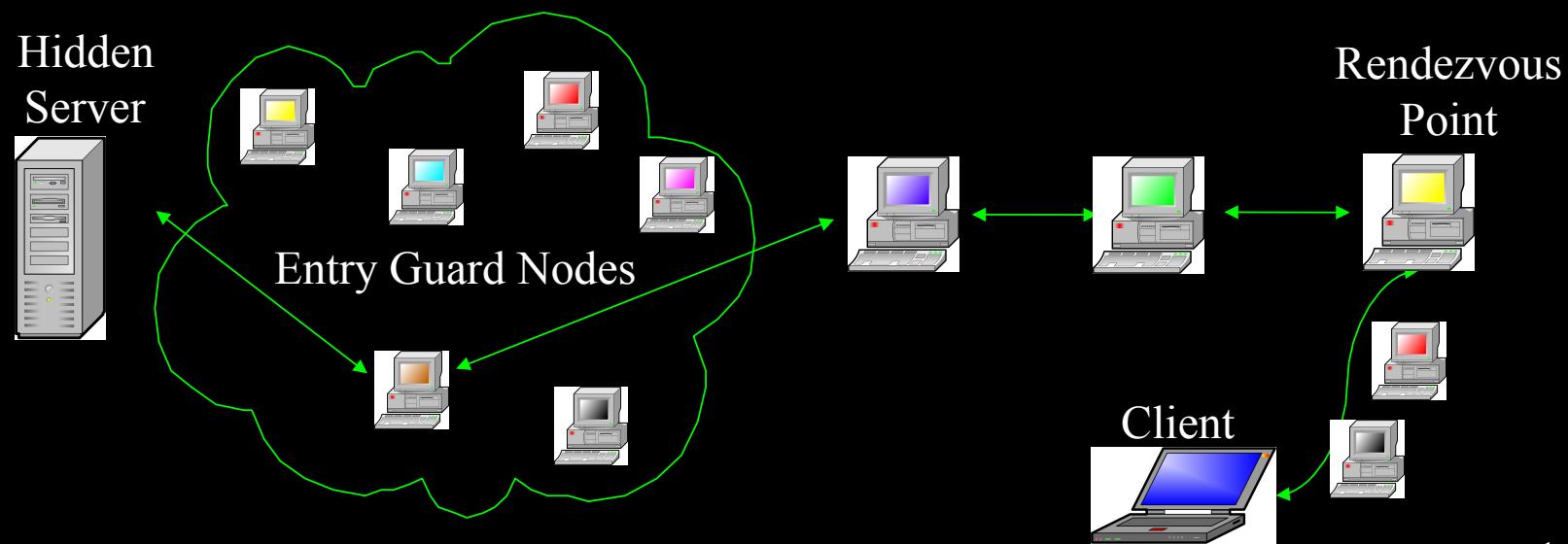
Dummy Traffic

- (Padding)
 - Often suggested
 - Expensive
 - Does **not** resist active attacks in low latency systems
- Easy to enforce a timing signature when inside a path



Entry Guard Nodes

- All first connections from Hidden Service are done through the same set of “guard” nodes
- Attacker may be running “old trusted/reliable nodes”!
- Will help against (but not eliminate!) the described attacks
- How to select nodes and determine size of set?
 - Random vs. Friend, Layered Entry Guards



Entry Guard Nodes and Predecessor Attacks

Such predecessor attacks were already known from prior work (cf. yesterday)

What we demonstrated was that these attacks were

- Significant
- Fast
- Cheap (owning a single node in the network)

Since one would lose anonymity so quickly and often anyway, using guards would mean you either lose immediately and always or never.

- Version of the general idea for various anonymous comms systems of “helper nodes”. (Wright et al. 2003)

Entry Guard Nodes and Predecessor Attacks

Attacks focused on what could be done using a single hostile node

With multiple adversary nodes, attacks should apply to general Tor circuits

- Bauer et al. (WPES'07) showed this to be true in simulation

We also showed that entry guards themselves easily identifiable by the same techniques

Next up: how best to choose guard nodes if you trust some nodes more than others.

What's next

Lecture 3:

- Formalization and analysis, possibilistic and probabilistic definitions of anonymity
- Hidden services: responder anonymity, predecessor attacks revisited, guard nodes

Lecture 4:

- Link attacks
- Trust

Questions?