Provable Unlinkability Against Traffic Analysis

Marcin Gomułkiewicz, Marek Klonowski and Mirek Kutyłowski

Wrocław University of Technology, Poland

ISC'2004

Anonymous communication

- a valuable information is **who is communicating with whom**
- hard to hide it in public networks!

Naive solution – all-to-all: send an encrypted message to all participants, keep sending even if no message need to be sent
communication overhead!

Onions

- generic, scalable technique for distributed systems,
- Rackoff and Simon ’91,
  re-invented: BABEL, ONION ROUTING 1996
  a kernel of TOR 2004

Onions

If A wants send a message m to server B

- A chooses at random λ intermediate nodes $J_1, \ldots, J_\lambda$;

Onions

If A wants send a message m to server B

- A chooses at random λ intermediate nodes $J_1, \ldots, J_\lambda$;
- A creates an onion:

$O :=$

$$\mathrm{Enc}_B(m)$$

Onions

If A wants send a message m to server B

- A chooses at random λ intermediate nodes $J_1, \ldots, J_\lambda$;
- A creates an onion:
  $O :=$

$$\mathrm{Enc}_{J_\lambda}(\mathrm{Enc}_B(m), B)$$

Onions

If A wants send a message m to server B

- A chooses at random λ intermediate nodes $J_1, \ldots, J_\lambda$;
- A creates an onion:

  $O :=$

$$\mathrm{Enc}_{J_{\lambda-1}}(\mathrm{Enc}_{J_\lambda}(\mathrm{Enc}_B(m), B), J_\lambda)$$

Onions

If A wants send a message m to server B

- A chooses at random λ intermediate nodes $J_1, \ldots, J_\lambda$;
- A creates an onion:
  $O :=$
  $\mathrm{Enc}_{J_1}(\ldots(\mathrm{Enc}_{J_{\lambda-1}}(\mathrm{Enc}_{J_\lambda}(\mathrm{Enc}_B(m), B), J_\lambda), J_{\lambda-1})\ldots, J_2)$.

Processing an Onion

If A wants send a message m encrypted as O to server B

- A sends onion O to J_1

Processing an Onion

If A wants send a message m encrypted as O to server B

- A sends onion O to J_1
- J_1 decrypts O and obtains some (O', J_2)

Processing an Onion

If A wants send a message m encrypted as O to server B

- A sends onion O to J_1
- J_1 decrypts O and obtains some (O', J_2)
- J_1 sends O' to J_2

Processing an Onion

If A wants send a message m encrypted as O to server B

- A sends onion O to J_1
- J_1 decrypts O and obtains some (O', J_2)
- J_1 sends O' to J_2
- J_2 decrypts ..
- J_2 sends .. to J_3

Processing an Onion

If A wants send a message m encrypted as O to server B

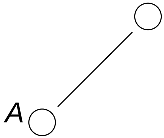
- A sends onion O to J_1
- J_1 decrypts O and obtains some (O', J_2)
- J_1 sends O' to J_2
- J_2 decrypts ..
- J_2 sends .. to J_3
- ...

Route of an onion

single onion

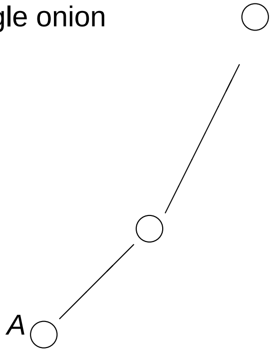
Route of an onion

single onion



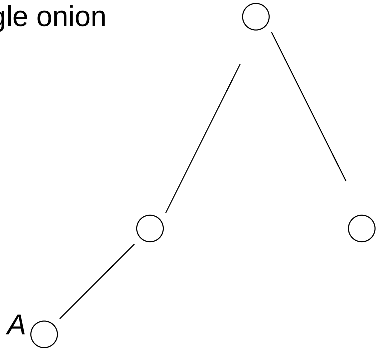
Route of an onion

single onion



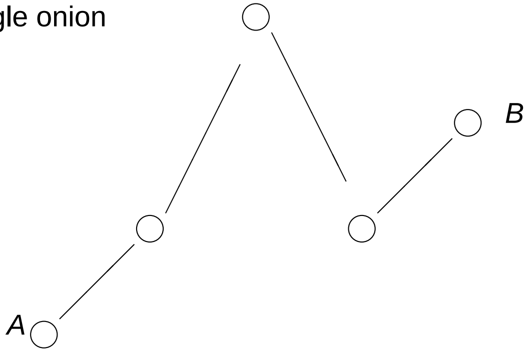
Route of an onion

single onion



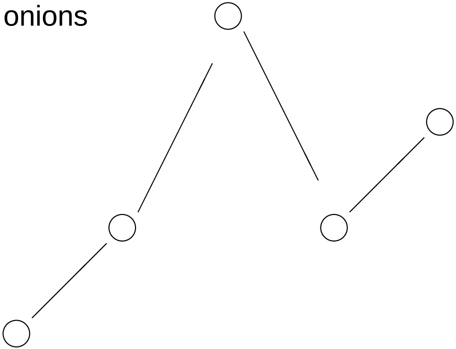
Route of an onion

single onion



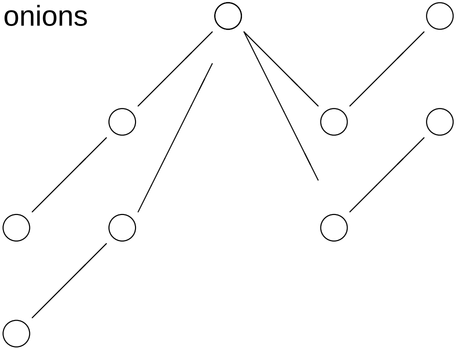
Onions at work

many onions



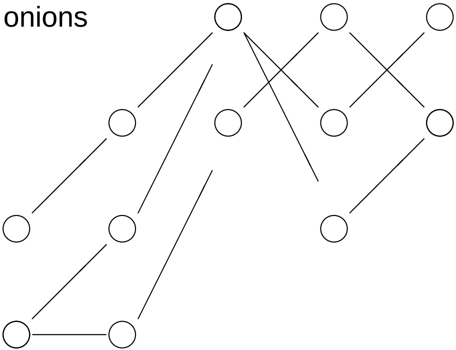
Onions at work

many onions



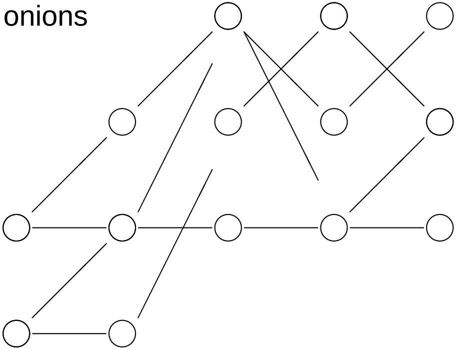
Onions at work

many onions



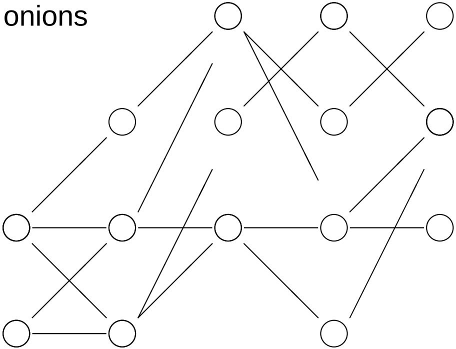
Onions at work

many onions



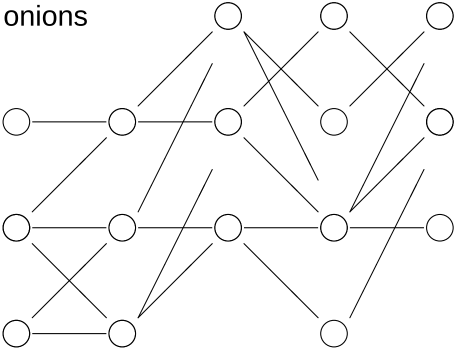
Onions at work

many onions



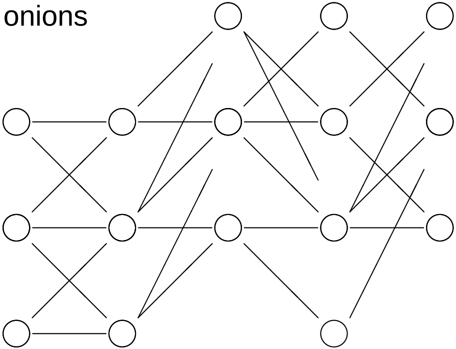
Onions at work

many onions



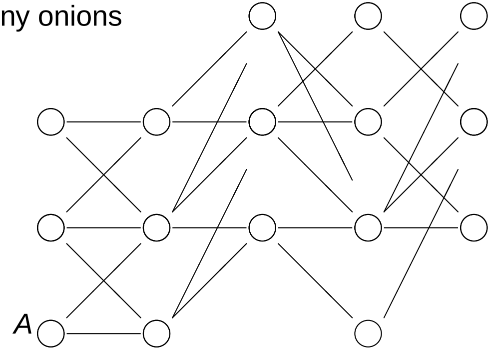
Onions at work

many onions



Onions at work

many onions



destination of the message starting at *A*?

Path length

- intuitively clear: anonymity level grows with growth of λ
- crucial question: how large must be λ in order to guarantee a solid anonymity level?

Viewpoint of an external observer

- no relationship can be derived between messages entering a node and leaving a node at the same time (probabilistic encryption has to be used)

Viewpoint of an external observer

- no relationship can be derived between messages entering a node and leaving a node at the same time (probabilistic encryption has to be used)
- but: transmitting a message from a node to another node can be detected

Traffic analysis

- an adversary tries to determine who is communicating with whom
  - without breaking cryptographic encoding, but
  - with some knowledge about the traffic

What is a “good anonymity level”

goal of an adversary: consider probability of each mapping between the origin nodes and the destination nodes

- attack succeeds, if the probabilities are skewed

What is a “good anonymity level”

goal of an adversary: consider probability of each mapping between the origin nodes and the destination nodes

- attack succeeds, if the probabilities are skewed
- if traffic information does not influence these probabilities substantially, then the traffic does not leak a substantial amount of information

What is a “good anonymity level”

goal of an adversary: consider probability of each mapping between the origin nodes and the destination nodes

- attack succeeds, if the probabilities are skewed
- if traffic information does not influence these probabilities substantially, then the traffic does not leak a substantial amount of information

attacks in practice: much smaller probability spaces

but: **we would like to show that no statistical analysis can succeed**

Why considering the whole mapping is important?

Important case - electronic elections

- Eve analyses the votes, and derives probabilities that Alice voted for X, for each single X
- if probability distribution is close to uniform, then the scheme is often told to preserve anonymity.

Why considering the whole mapping is important?

Important case - electronic elections

- Eve analyses the votes, and derives probabilities that Alice voted for X, for each single X
- if probability distribution is close to uniform, then the scheme is often told to preserve anonymity.

FALSE!

- Eve may be unable to derive preferences of Alice

Why considering the whole mapping is important?

Important case - electronic elections

- Eve analyses the votes, and derives probabilities that Alice voted for X, for each single X
- if probability distribution is close to uniform, then the scheme is often told to preserve anonymity.

FALSE!

- Eve may be unable to derive preferences of Alice
- but can deduce that Alice and Bob voted for the same party with probability 90%

Adversaries

passive adversary :

model 1 an adversary can monitor the whole traffic

model 2 only a fraction of connections may be traced at each moment

Adversaries

passive adversary :

- model 1: an adversary can monitor the whole traffic
- model 2: only a fraction of connections may be traced at each moment

active adversary : may influence the traffic

- non-adaptive: an attack cannot be adapted to the traffic observed
- adaptive

Security proofs for onions - results

assumptions: **passive adversary, 1 packet messages, onion paths of length λ.**

An adversary can monitor the whole traffic:

- no security proof for the original protocol
- modified version of the protocol (routing in growing groups)
  Rackoff, Simon, FOCS'91, for $\lambda \approx \log^{11} n$,
  Czumaj, Kanarek, Kutyłowski, Loryś, SODA'98, for $\lambda = O(\log^2 n)$

Only a fraction of connections may be traced:

- Berman, Fiat, Ta-Shma, FC'2004, for $\lambda = O(\log^4 n)$

This presentation: **for** $\lambda = \Theta(\log n)$

Traffic analysis - assumptions

- an adversary can see
  - all messages sent at source nodes
  - all messages received by destination nodes
- cryptographic encoding ensures that only **the number of messages** can be detected, no other information leaked
- an adversary can see the number of messages transmitted at the links (determined by the adversary in advance)
- a constant fraction of links can be traced (not necessarily the same all the time)

Outcome of Traffic Analysis

- random variable π:
  $\pi(i) = j$ iff the ith message is delivered at the jth delivery point
- a priori probability: $\Pr(\pi)$ – known by an adversary
- traffic information yields conditioned probabilities:

$$\Pr(\pi | C)$$

  where C is the observed traffic
  (for instance a lack of a path may betray that $\pi(i) \neq j$ with probability 1)

Protocol Immune to Traffic Analysis

- probability distributions $\Pr(\pi)$ and $\Pr(\pi|C)$ do not differ substantially

Protocol Immune to Traffic Analysis

- probability distributions $\Pr(\pi)$ and $\Pr(\pi|C)$ do not differ substantially
- for some C traffic analysis for onion protocol reveals everything: i.e. if the paths of messages are disjoint
- goal: show that

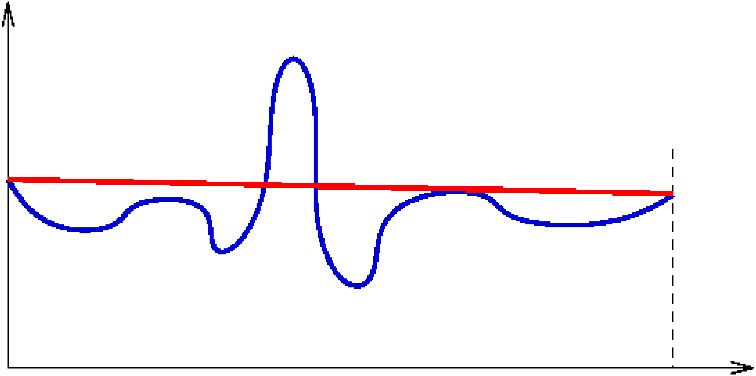
$$\Pr(\pi) \approx \Pr(\pi|C)$$

for almost all C

Variation distance

The total variation distance between probability distributions μ_1 and μ_2 defined over space X of elementary events equals

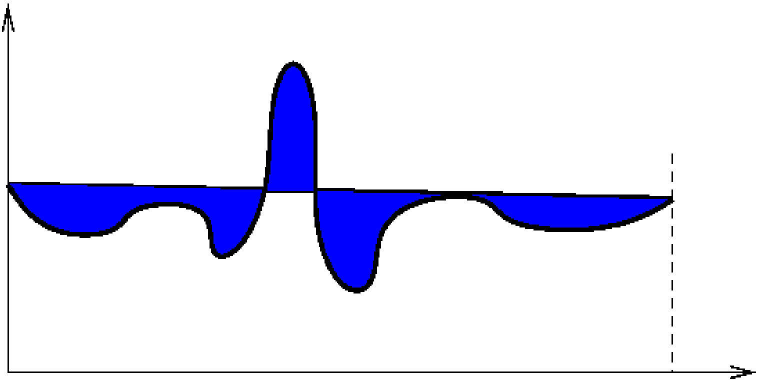
$$\|\mu_1 - \mu_2\| = \frac{1}{2} \sum_{x \in X} |\mu_1(x) - \mu_2(x)| \ .$$



Variation distance

The total variation distance between probability distributions μ_1 and μ_2 defined over space X of elementary events equals

$$\|\mu_1 - \mu_2\| = \frac{1}{2} \sum_{x \in X} |\mu_1(x) - \mu_2(x)| \ .$$



Simplified case

- for each user: uniform probability distribution over destination points
- Berman, Fiat, Ta-Shma show how to generalize the results to non-uniform distributions (FC'2004)

Sending messages as a stochastic process

- at each step the messages are sent to next locations at random
- but so that the traffic adheres to the traffic observed by an adversary
  for simplicity assume that the adversary can see the number of messages at **each node**

Stationary distribution

- a probability distribution over the set of states is **stationary** if applying a single step of the process does not change the probability distribution,

Stationary distribution

- a probability distribution over the set of states is **stationary** if applying a single step of the process does not change the probability distribution,
- in our case: a uniform distribution of messages 1 through m over m locations holding messages

How many steps are needed until probability distribution becames close to the uniform distribution?

Rapid mixing techniques

Goal:

- given a stochastic process $\mathcal{P}$ with a stationary distribution u
- show that after t steps the probability distribution of the process started in an arbitrary state is close to u

Rapid mixing techniques

Goal:

- given a stochastic process $\mathcal{P}$ with a stationary distribution u
- show that after t steps the probability distribution of the process started in an arbitrary state is close to u

How to construct such a proof?

Coupling techniques

- define two processes $\mathcal{P}_A, \mathcal{P}_B$
- both are the copies of $\mathcal{P}$,

Coupling techniques

- define two processes $\mathcal{P}_A, \mathcal{P}_B$
- both are the copies of $\mathcal{P}$,
- but the choices of the first process may influence the second process

Coupling goal

- define dependencies so that the processes “converge” – (with probabilities growing with the number of steps) they reach the same state

Coupling goal

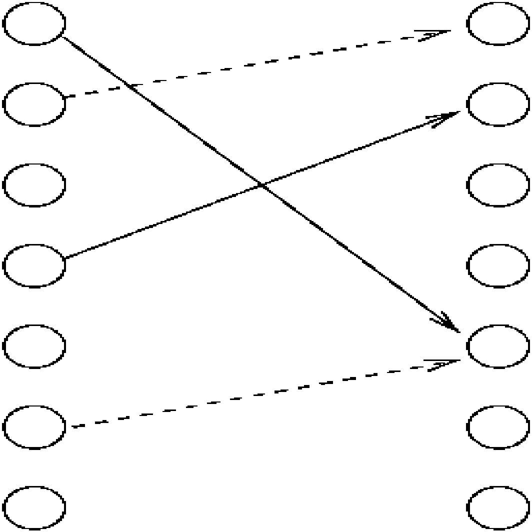
- define dependencies so that the processes “converge” – (with probabilities growing with the number of steps) they reach the same state
- key property – coupling lemma:

$$\text{variation distance after } t \text{ steps} \\ \leq \\ \mathsf{Pr}[\mathcal{P}_A \text{ and } \mathcal{P}_B \text{ differ after } t \text{ steps}].$$

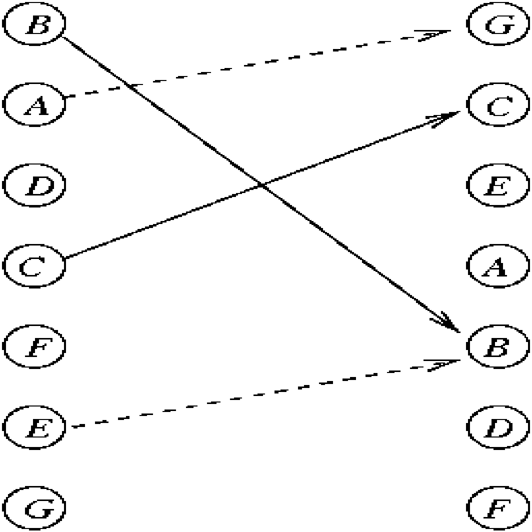
Path coupling

- it suffices to consider processes that are almost in the same state
  - distance function between process states; values $1, 2, \ldots$, for each pair of states a "path" where neighbors are at distance 1,
  - it suffices to consider pair of processes at distance 1

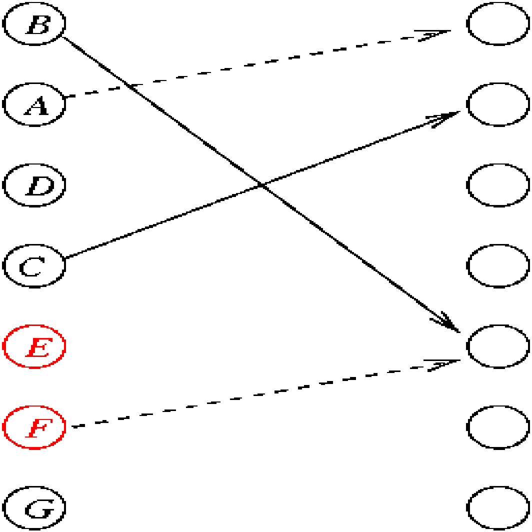
Coupling rule - traffic information



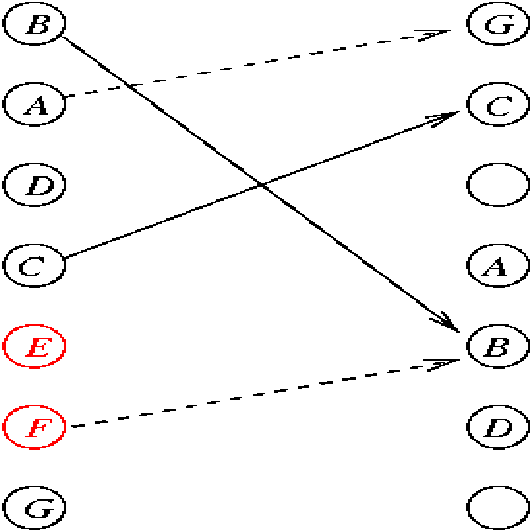
Coupling rule - transition of process I



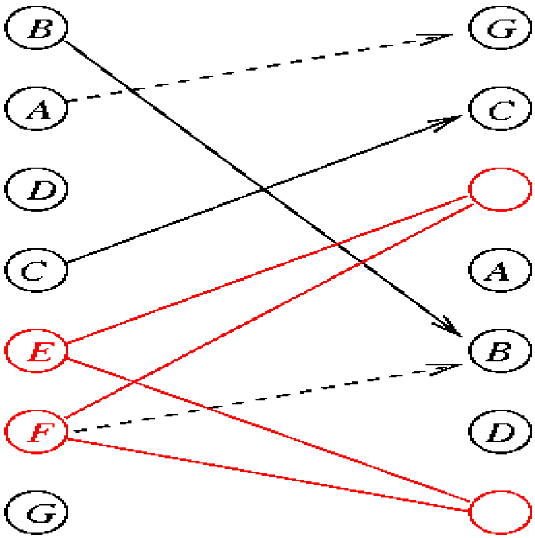
Coupling rule - state of process II



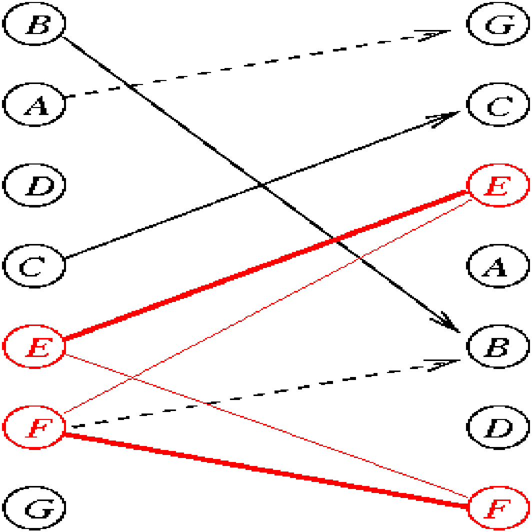
Coupling rule - transition of process II



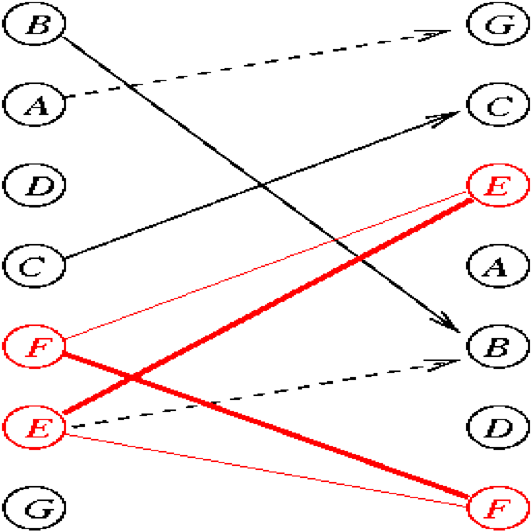
Coupling rule - crossover



Coupling rule - transition of process II



Coupling rule - transition of process I



Path coupling

- large number of crossovers regardless of the strategy of an adversary (Lemma of Noga Alon)
- 2 steps – processes couple with probability $\geq$ *const*

Remarks and Conclusions

- somewhat strange technique but: **strong and easy to use**
- coupling proofs also work well for “limited anonymity” targets
- other results:
  - on Chaum’s electronic voting scheme (2003)
  - on networks of mixes (2004?)

Thanks for your attention!