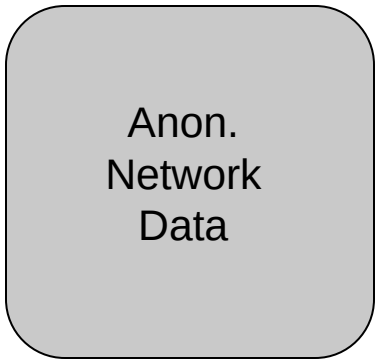
Taming the Devil:
Techniques for Evaluating Anonymized Network Data

Scott Coull[1], Charles Wright[1], Angelos Keromytis[2],
Fabian Monrose[1], Michael Reiter[3]

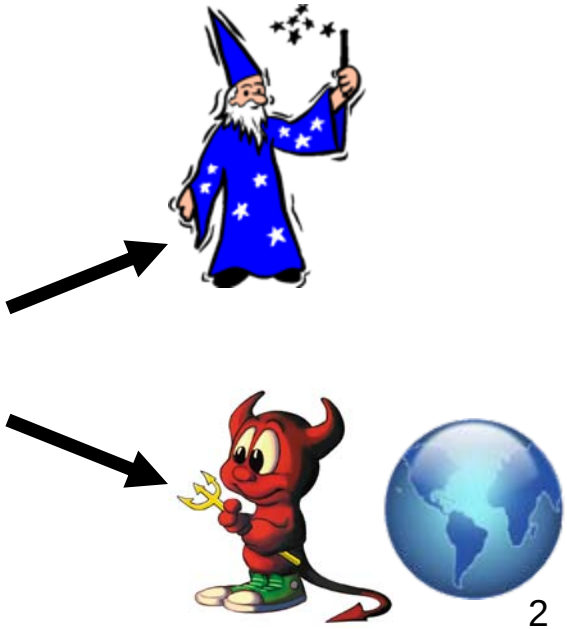
Johns Hopkins University[1] Columbia University[2]
University of North Carolina - Chapel Hill[3]

The Network Data Sanitization Problem

- Anonymize a packet trace or flow log s.t.:
  1. Researchers gain maximum utility
  2. Adversaries w/ auxiliary information do not learn sensitive information



Anonymization



Methods of Sanitization

- Pseudonyms for IPs
  - Strict prefix-preserving [FXAM04]
  - Partial prefix-preserving [PAPL06]
  - Transaction-specific [OBA05]

- Other data fields anonymized in **reaction** to attacks
  - *e.g.,* time stamps are quantized due to clock skew attack [KBC05]

Notable Attacks

- Several active and passive attacks exist…
  - Active probing [BA05, BAO05,KAA06]
  - Host profiling [CWCMR07,RCMT08]
  - Identifying web pages [KAA06, CCWMR07]

The Underlying Problem

- Attacks can be generalized as follows:
  1. Identifying information is encoded in the anonymized data
     - Host behaviors for profiling attacks
  2. Adversary has external information on true identities
     - Public information on services offered by a host
  3. Adversary maps true identities to pseudonyms

Our Goals

1. Find objects at risk of deanonymization
2. Compare anonymization systems and policies
3. Model hypothetical attack scenarios

- *Focus on ‘natural’ sources of information leakage*

Related Work

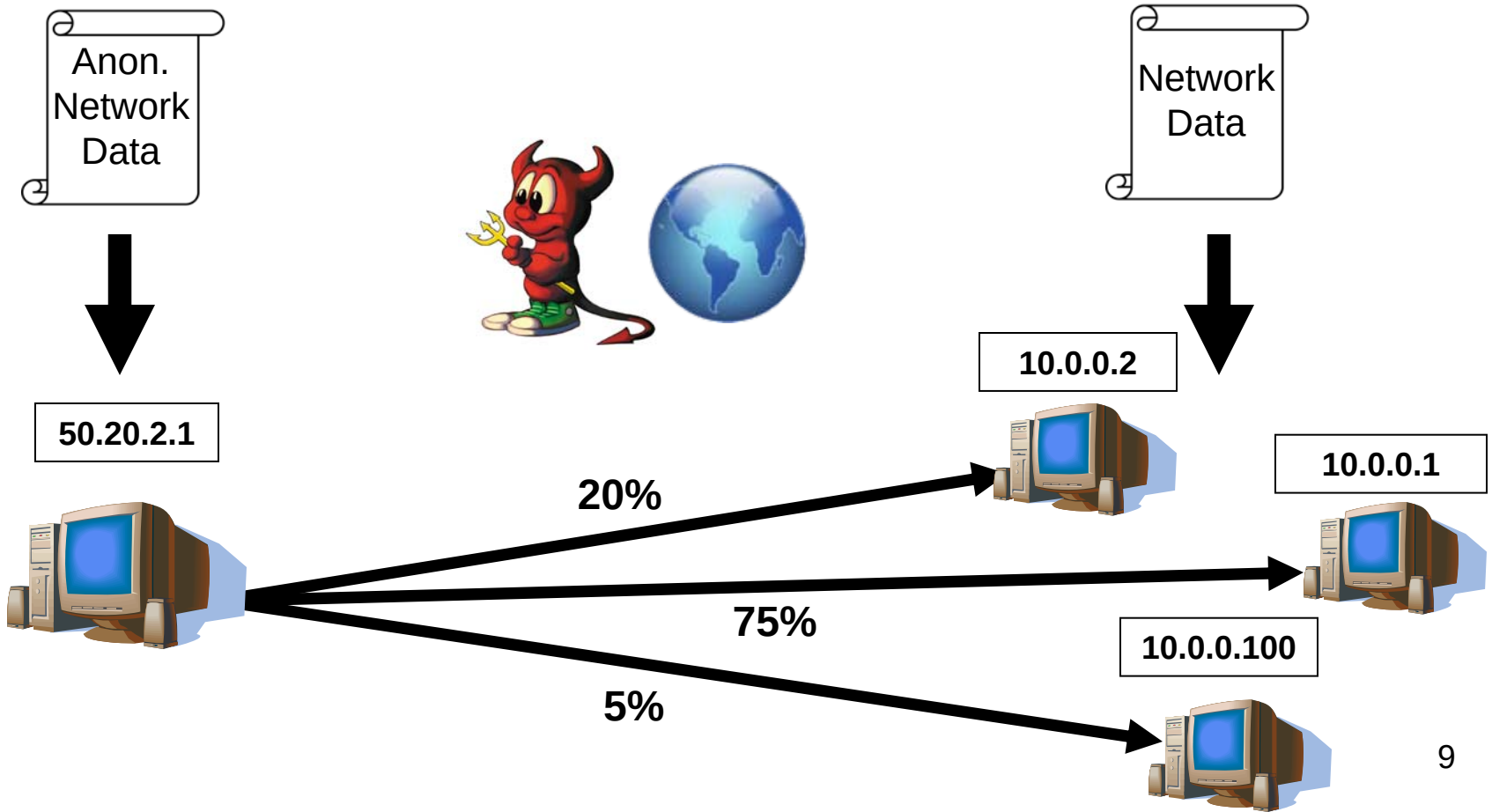
- Definitions of Anonymity
  - k-Anonymity [SS98], ℓ-Diversity [MGKV05], and t-Closeness[LLV07]
- Information theoretic metrics
  - Analysis of anonymity in mixnets [SD02][DSCP02]
- An orthogonal method for evaluating network data [RCMT08]

Outline

- Adversarial Model
- Defining Objects
- Auxiliary Information
- Calculating Anonymity
- Evaluation

Adversarial Model

- Adversary's goal: map an ***anonymized object*** to its ***unanonymized*** counterpart



Defining Objects

- Consider network data as a database
  - *n* rows, *m* columns
  - Each row is a packet (or flow) record
  - Each column is a data field (*e.g.,* source port)

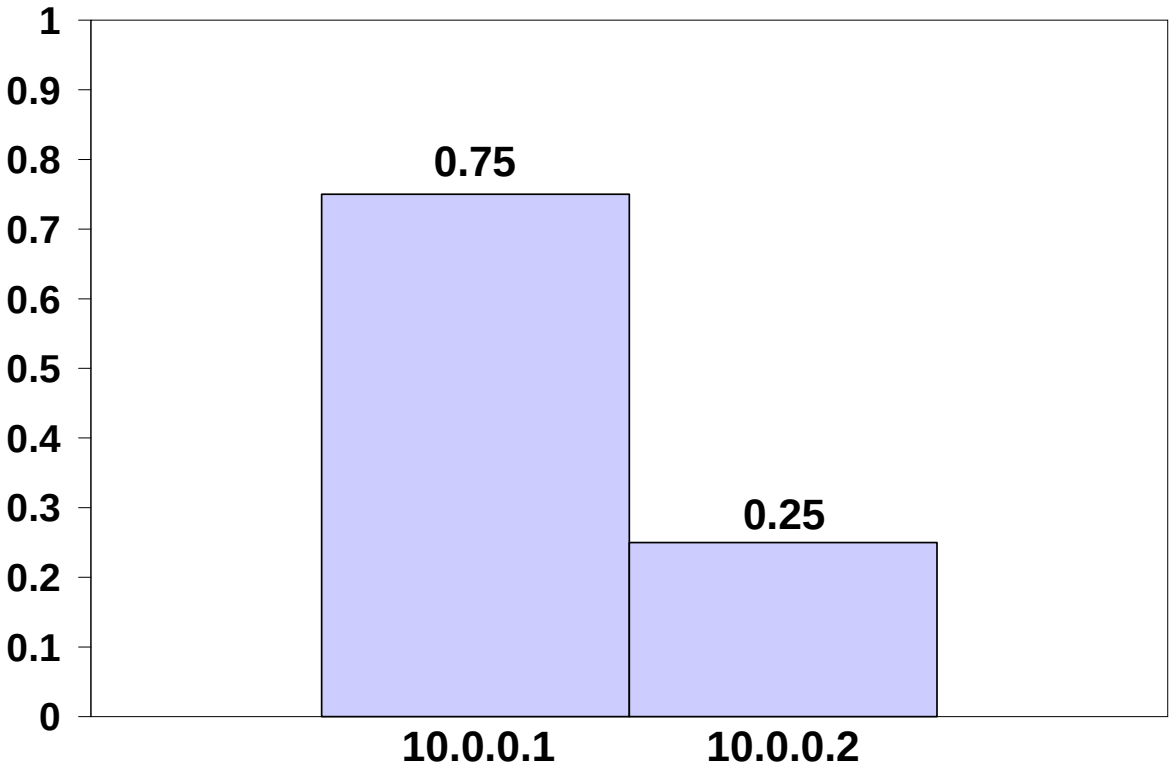
- Fields can induce a ***probability distribution***
  - Sample space defined by values in the field
  - Represented by random variables in our analysis

Defining Objects

ID	Local IP	Local Port	Remote IP	Remote Port
1	10.0.0.1	80	192.168.2.5	1052
2	10.0.0.2	3069	10.0.1.5	80
3	10.0.0.1	80	192.168.2.10	4059
4	10.0.0.1	21	192.168.6.11	5024
		...		

Defining Objects

Local IP
10.0.0.1
10.0.0.2
10.0.0.1
10.0.0.1
…



Defining Objects

- Combinations of fields can leak information even if the fields are indistinguishable in isolation
  - A real-world adversary has a directed plan of attack on a certain subset of fields
  - Our analysis must consider a much larger set of potential fields
- Use feature selection methods based on mutual information to find related fields
  - Limits computational requirements

Defining Objects

- A ***feature*** is a group of correlated fields
  - Calculate ***normalized mutual information***
  - Group into pairs if mutual information > *t*
  - Merge groups that share a field in to a feature

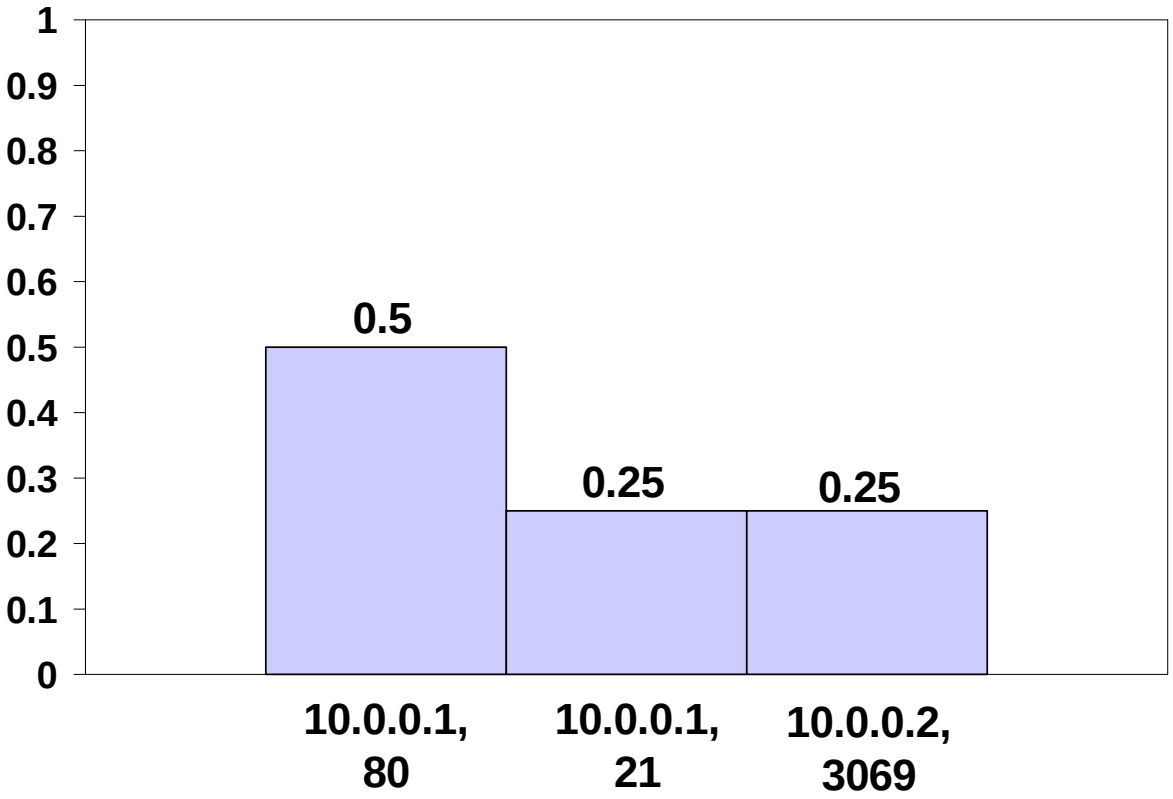
- A ***feature distribution*** is the joint distribution over the fields in the feature

Defining Objects

ID	Local IP	Local Port	Remote IP	Remote Port
1	10.0.0.1	80	192.168.2.5	1052
2	10.0.0.2	3069	10.0.1.5	80
3	10.0.0.1	80	192.168.2.10	4059
4	10.0.0.1	21	192.168.6.11	5024
		...		

Defining Objects

Local IP	Local Port
10.0.0.1	80
10.0.0.2	3069
10.0.0.1	80
10.0.0.1	21
…	



Defining objects

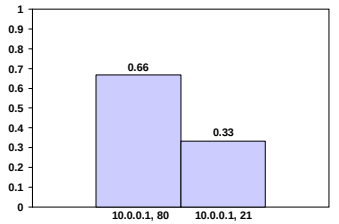
- An ***object*** is a set of feature distributions over records produced due its presence
  - *e.g.,* host objects – feature distributions induced by records sent from or received by a given host

Defining Objects

ID	Local IP	Local Port	Remote IP	Remote Port
1	10.0.0.1	80	192.168.2.5	1052
2	10.0.0.2	3069	10.0.1.5	80
3	10.0.0.1	80	192.168.2.10	4059
4	10.0.0.1	21	192.168.6.11	5024
...				

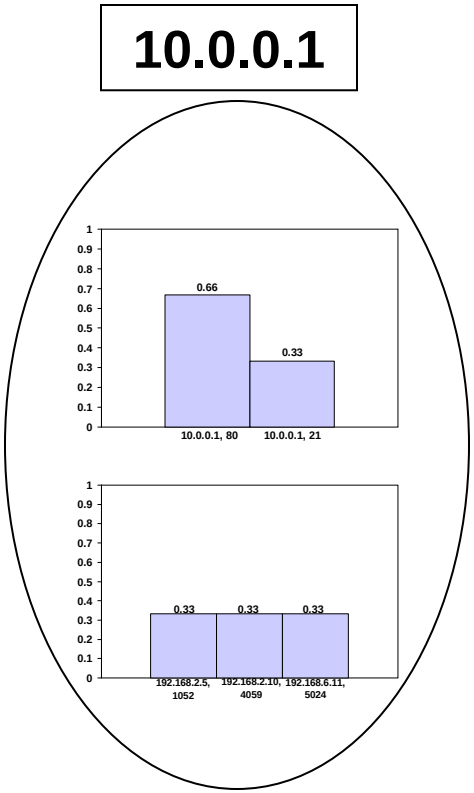
Defining Objects

ID	Local IP	Local Port	Remote IP	Remote Port
1	10.0.0.1	80	192.168.2.5	1052
3	10.0.0.1	80	192.168.2.10	4059
4	10.0.0.1	21	192.168.6.11	5024
		...		

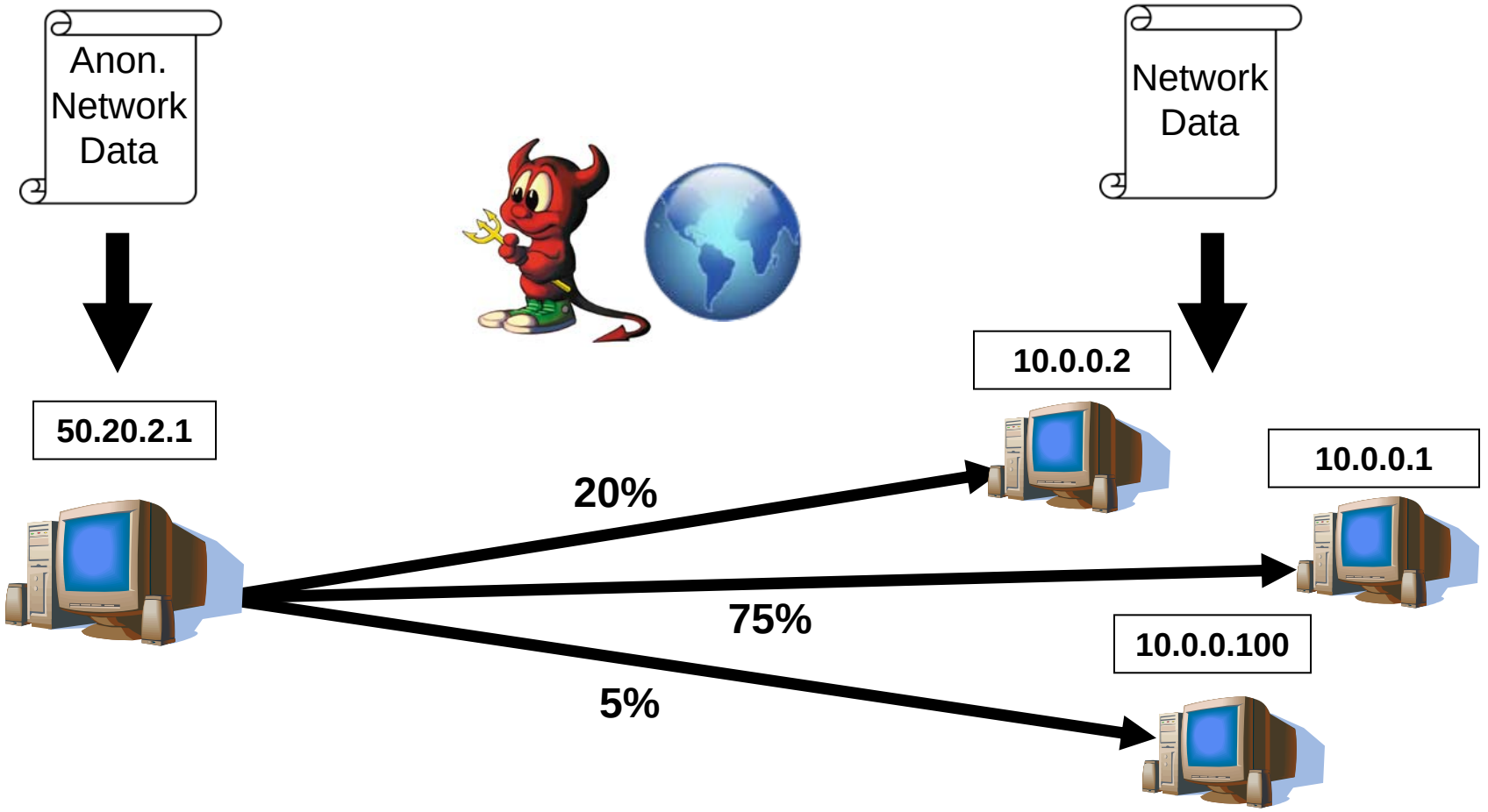


Defining Objects

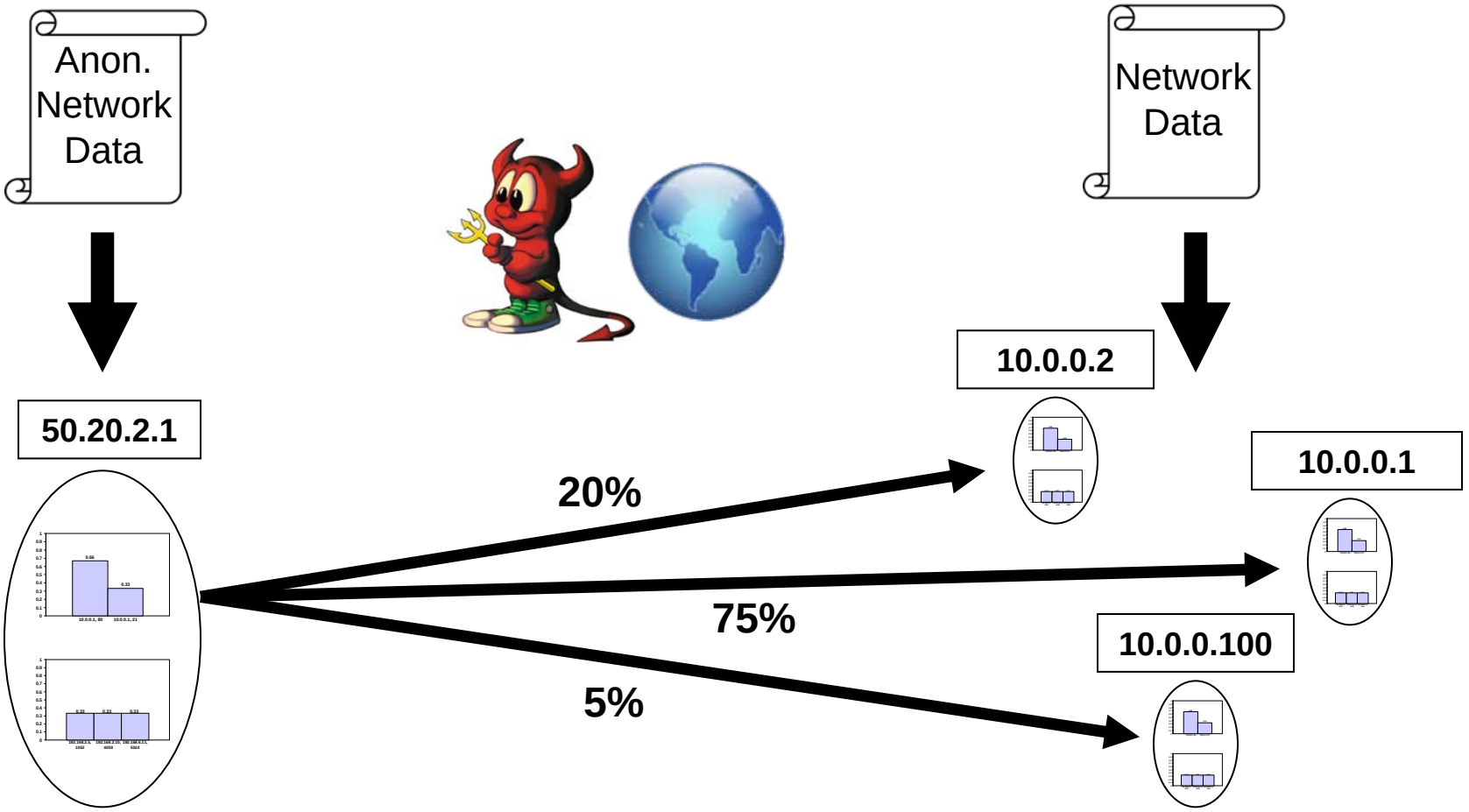
ID	Local IP	Local Port	Remote IP	Remote Port
1	10.0.0.1	80	192.168.2.5	1052
3	10.0.0.1	80	192.168.2.10	4059
4	10.0.0.1	21	192.168.6.11	5024
...				



Adversarial Model



Adversarial Model



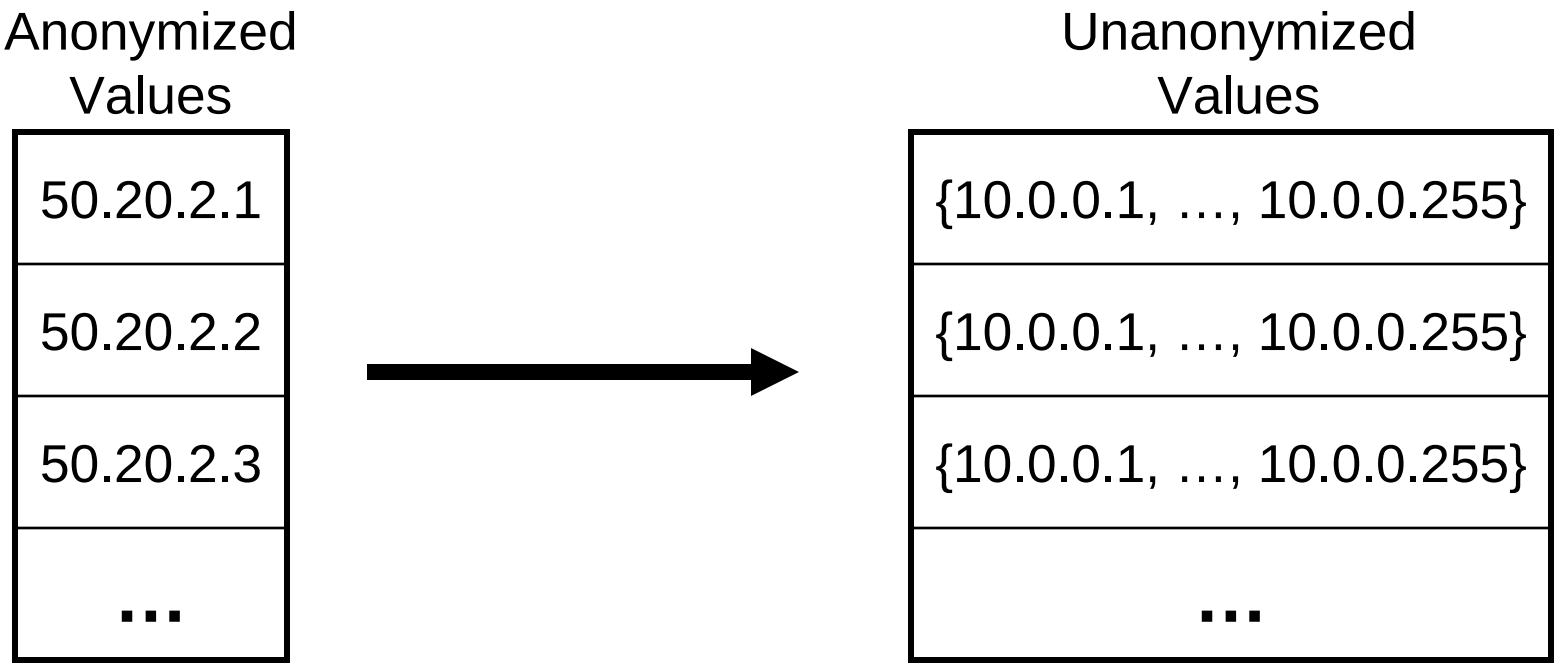
Auxiliary Information

- ***Auxiliary information*** captures the adversary's external knowledge
  - Initially, adversary only has knowledge obtained from meta-data
  - As adversary deanonymizes objects, new knowledge is gained
  - Used to iteratively refine mapping between anonymized and unanonymized objects

Auxiliary Information

Local IP:

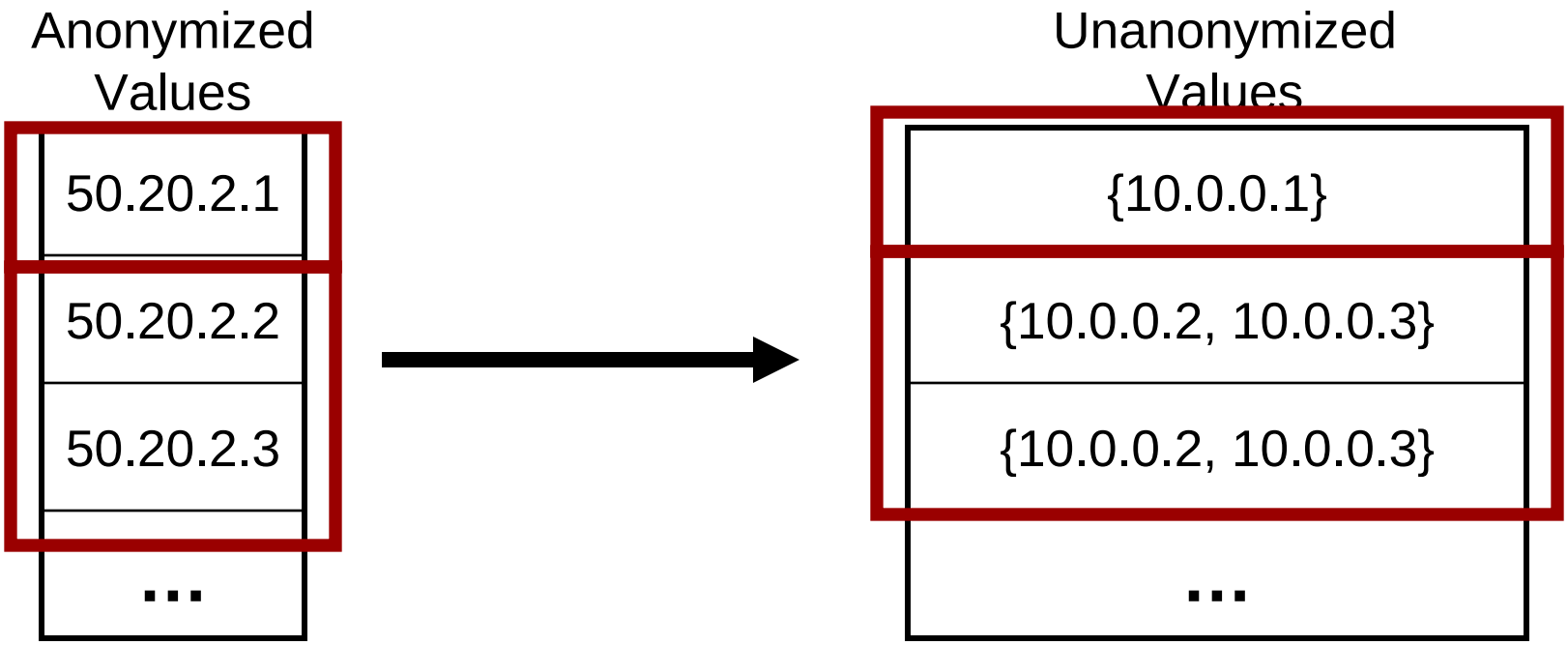
Prefix-Preserving



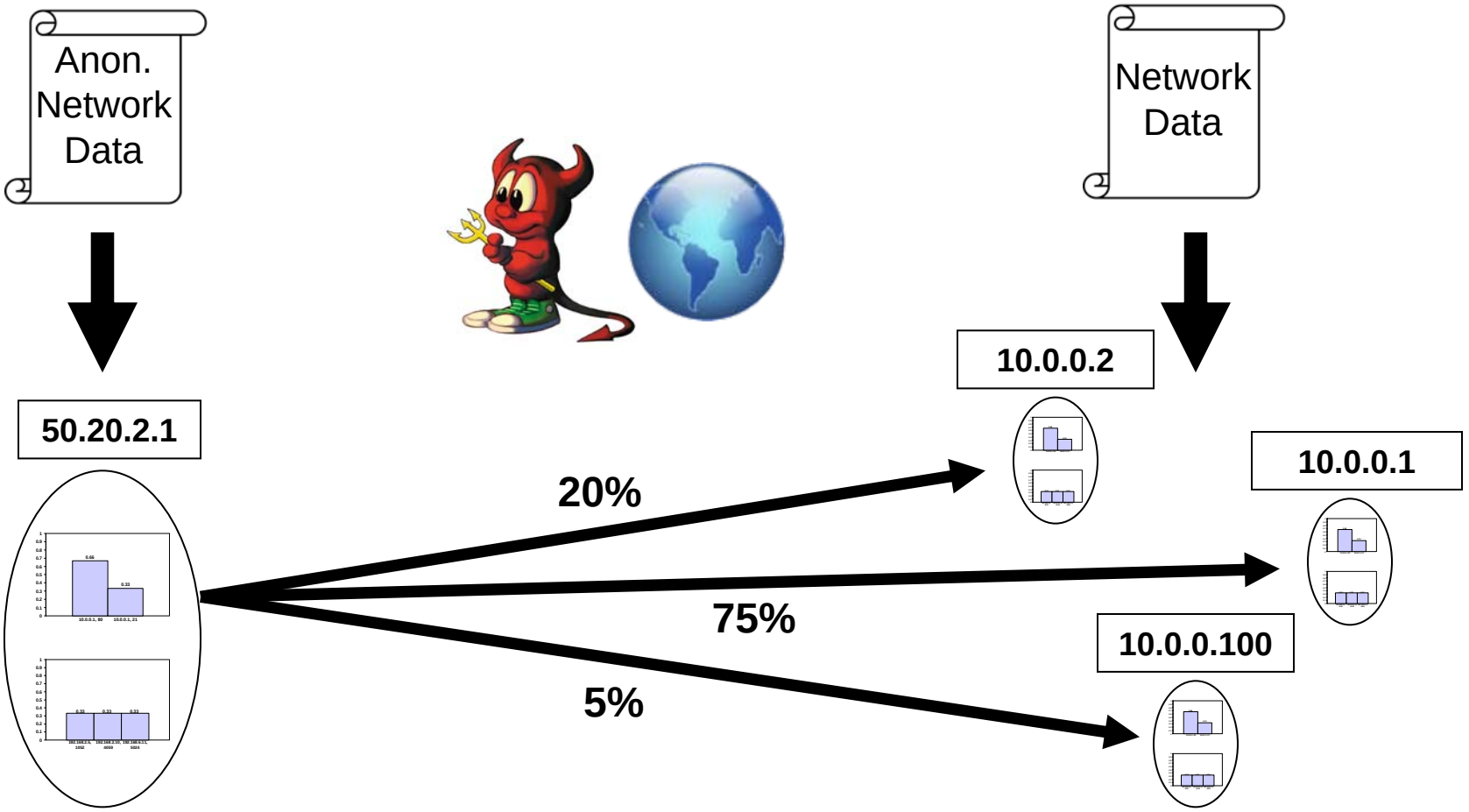
Auxiliary Information

Local IP:

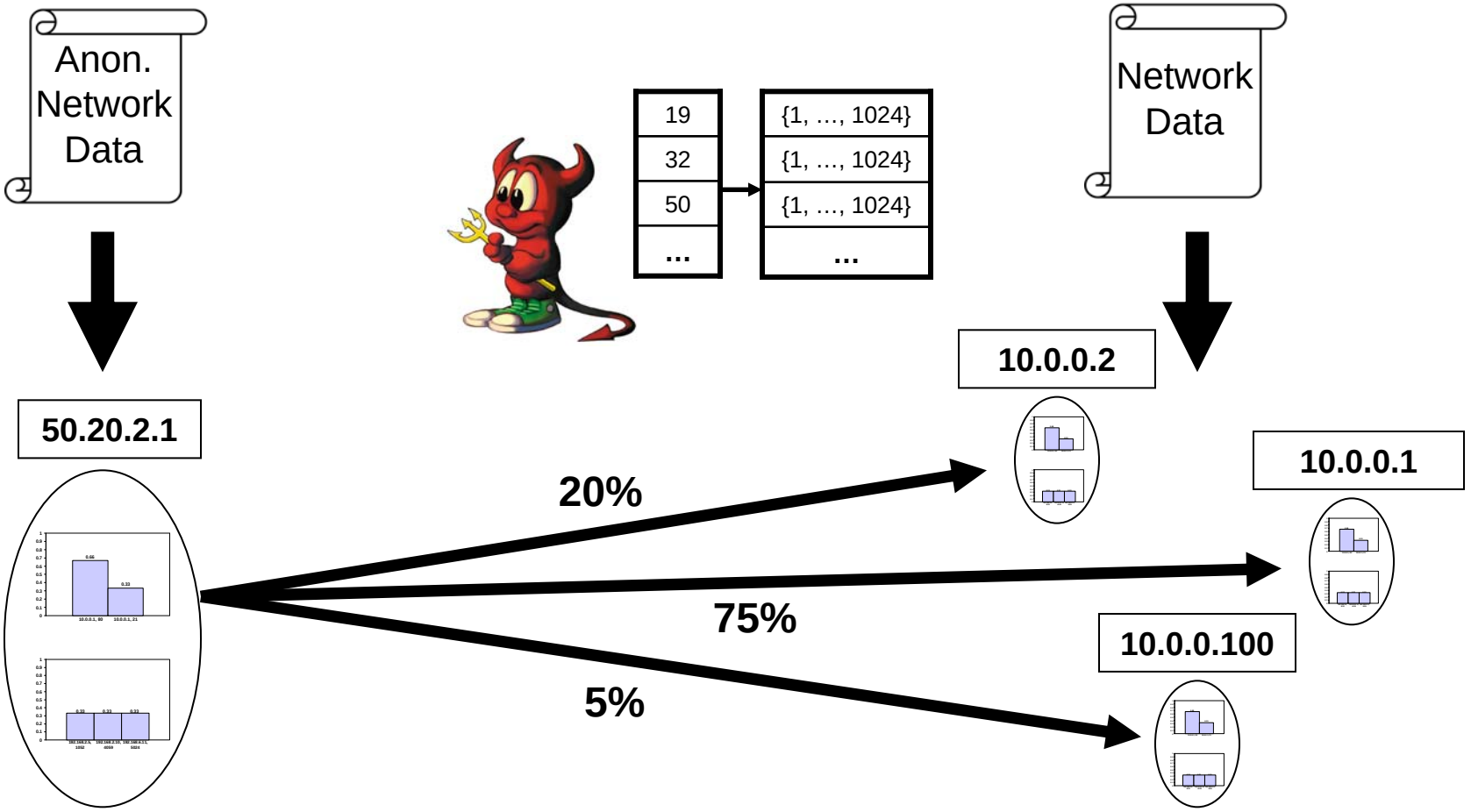
Prefix-Preserving



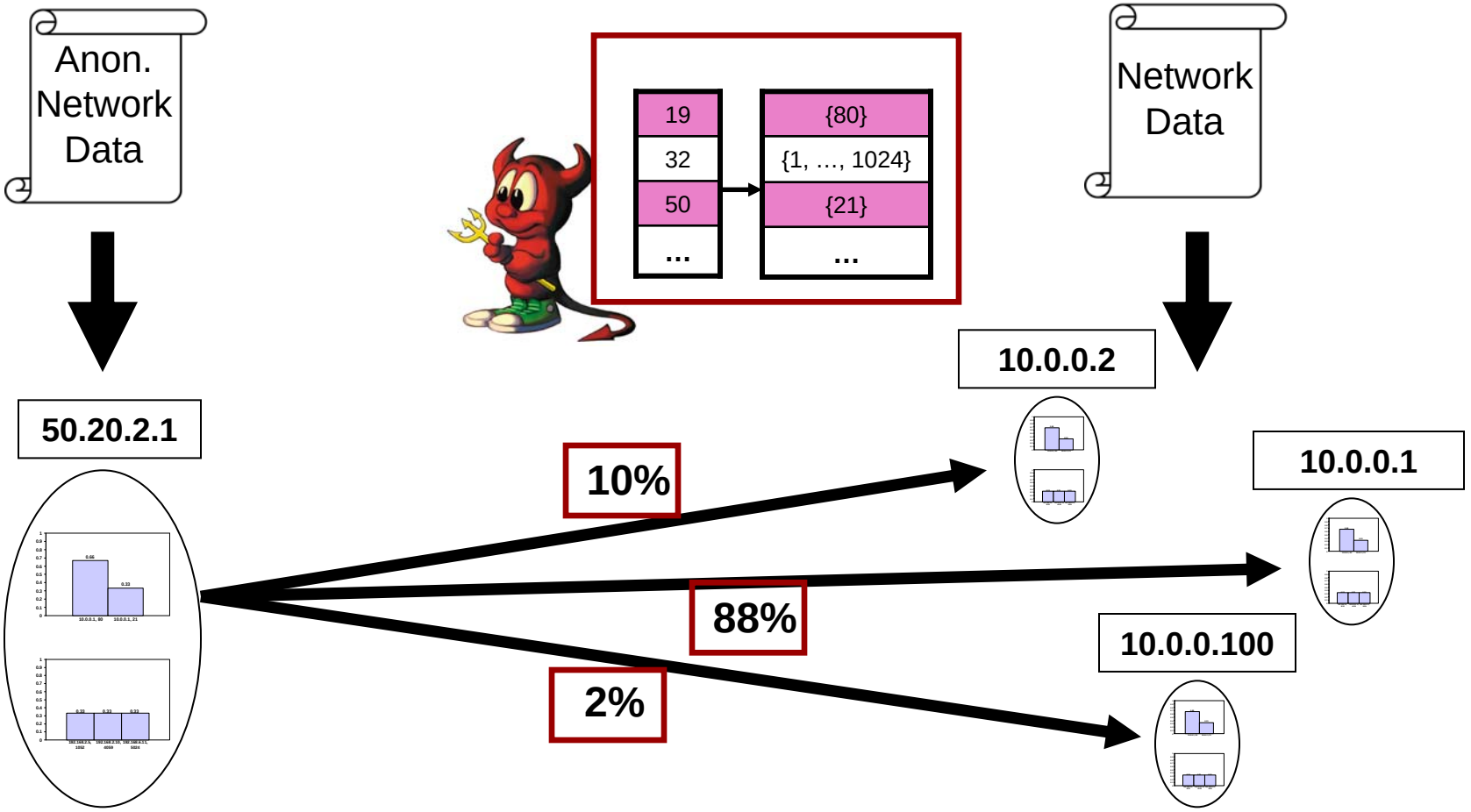
Adversarial Model



Adversarial Model



Adversarial Model



Calculating Anonymity

- Compare each feature distribution of anonymized object against all unanonymized objects
  - Use ***L1 similarity*** measure as a count to approximate a probability distribution
  - Use ***information entropy*** of the distribution as ***object anonymity*** with respect to the feature
- Auxiliary information dictates how the features are compared

Calculating Anonymity

- Sum of entropy across all features gives us the ***overall object anonymity***
  - Assuming features are independent due to mutual information correlation criterion

- Calculate ***conditional anonymity*** of an object via a greedy algorithm
  1. Choose lowest entropy object and assume it has been deanonymized
  2. Reverse anonymization to learn mappings
  3. Recalculate object anonymity with new auxiliary information

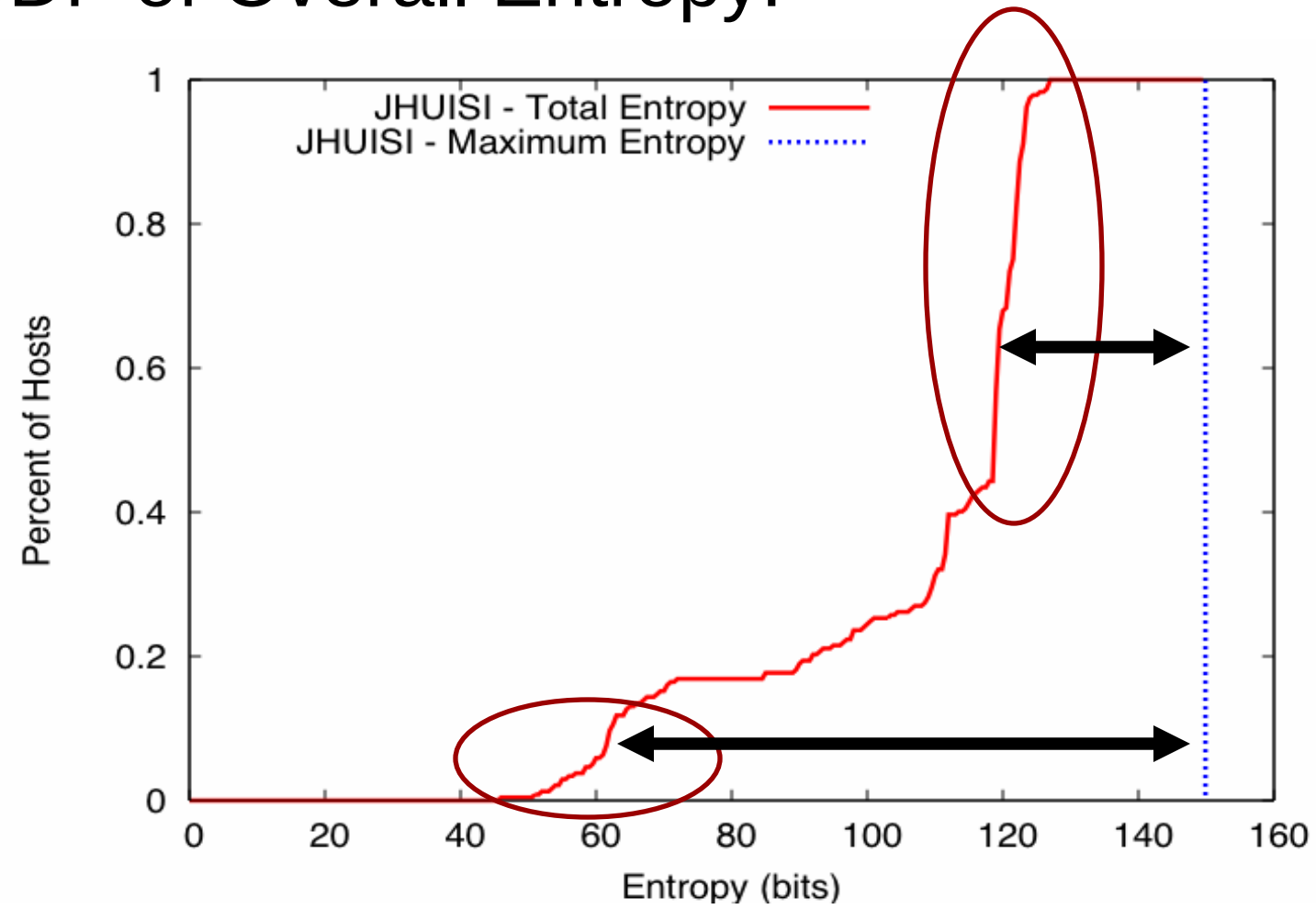
Evaluation

- Capture flow logs at the edge of the JHUISI network
  - 24 hours of data
  - 27,753 flows
  - 237 hosts on three subnets
  - Anonymized with *tcpmkpub* [PAPL06]

- Analysis of Host Objects
  - Defined by unique Local IPs
  - 19 features generated from the fields:
    - Start time, end time, local IP, local port, local size, remote IP, remote port, remote size, and protocol

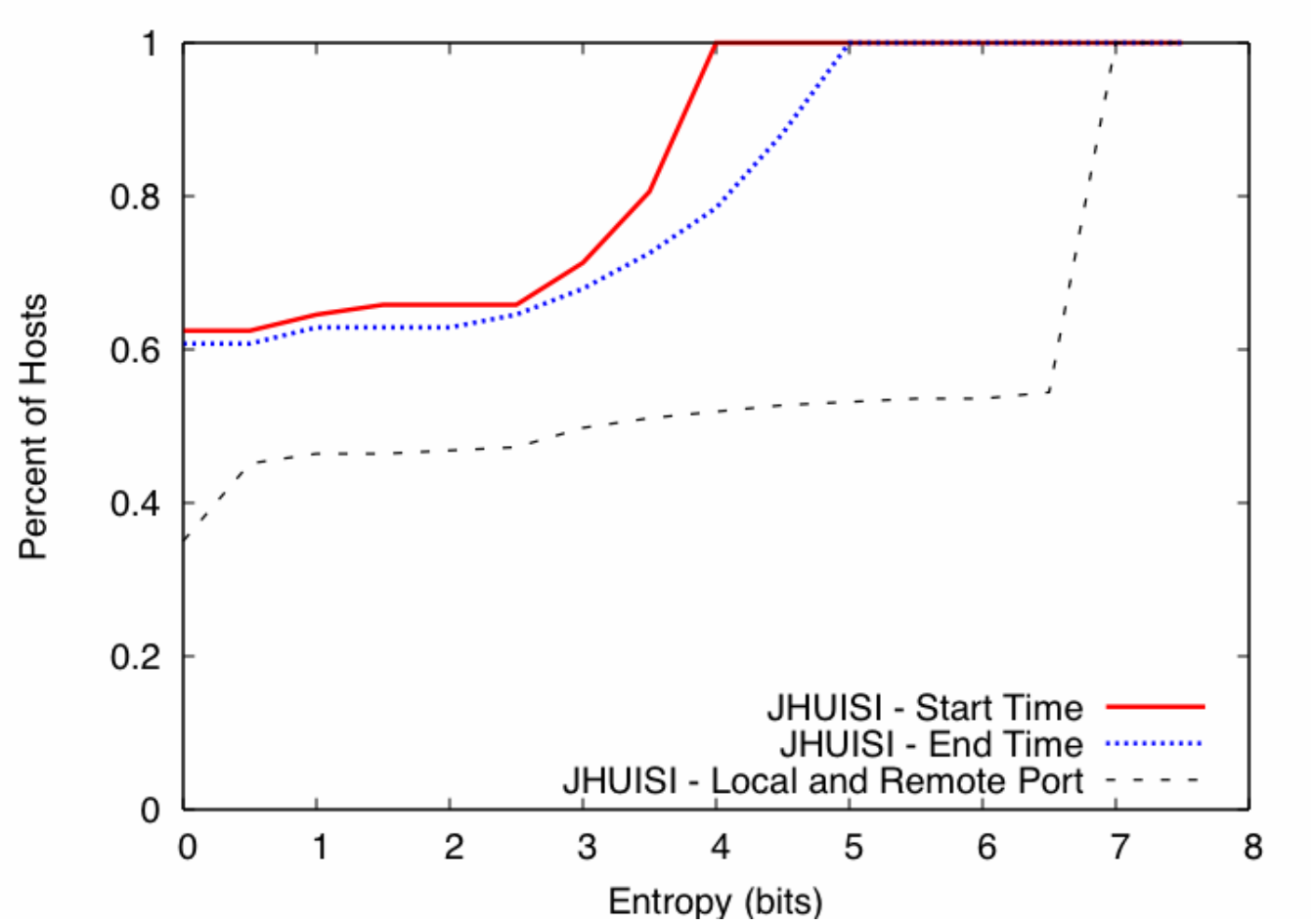
Evaluation

- CDF of Overall Entropy:



Evaluation

- CDF of three worst features:

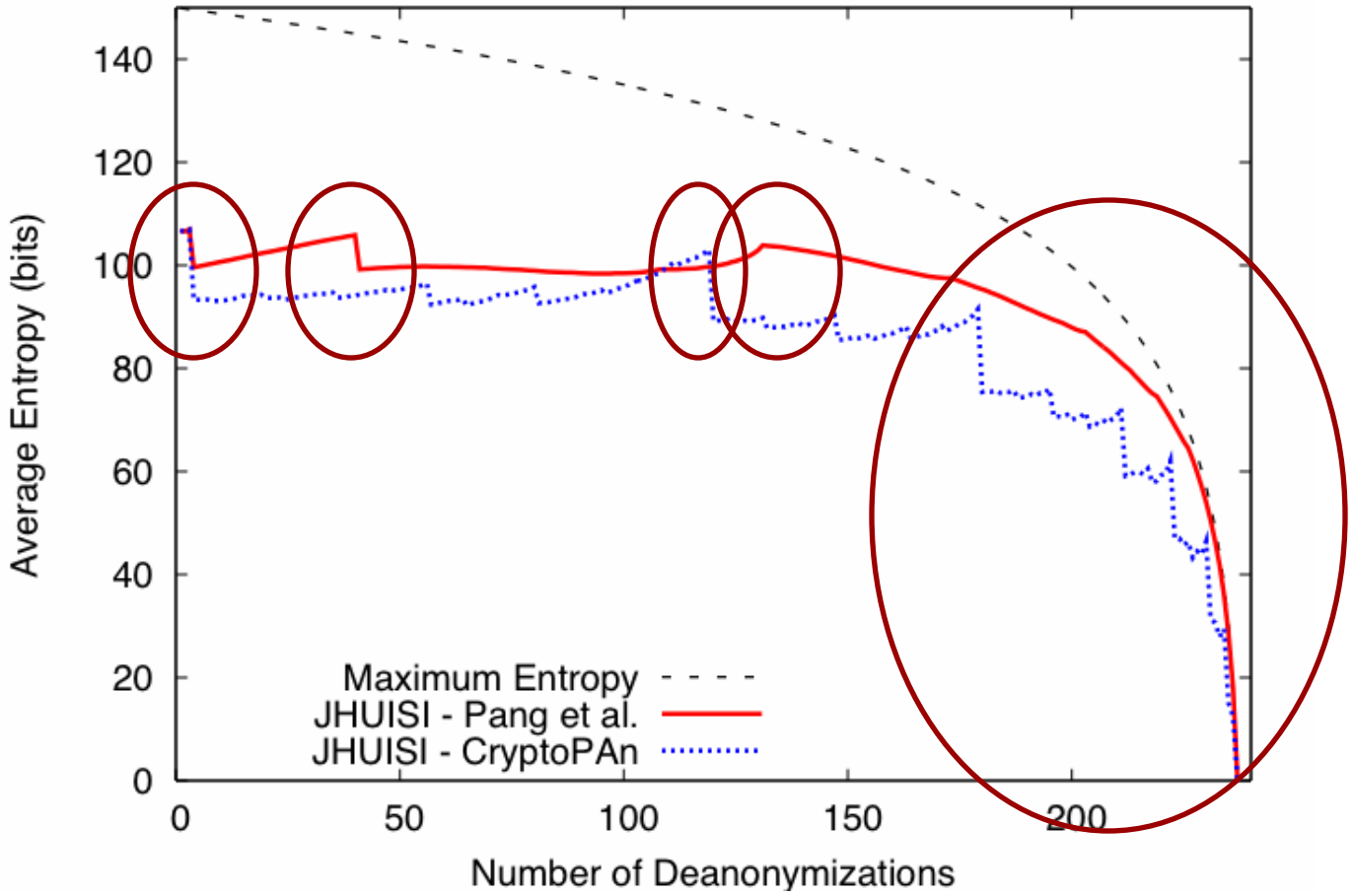


Evaluation

- Comparison of prefix-preserving schemes using conditional anonymity:
  - **CryptoPAn [FXAM04]** – if *n* bits of a prefix are shared in the unanonymized IP, *n* bits will be shared in the anonymized IP
  - **Pang *et al* [PAPL06]** – use pseudorandom permutation to anonymize host and subnet portions separately

Evaluation

- CryptoPAn vs. Pang *et al*:



Evaluation

- Conditional anonymity can also be used to evaluate the impact of known attacks
  - Simulate the behavioral profiling attack [CWCMR07]
  - Determine the hosts that are susceptible
  - Determine the impact of deanonymizing those hosts on those that remain

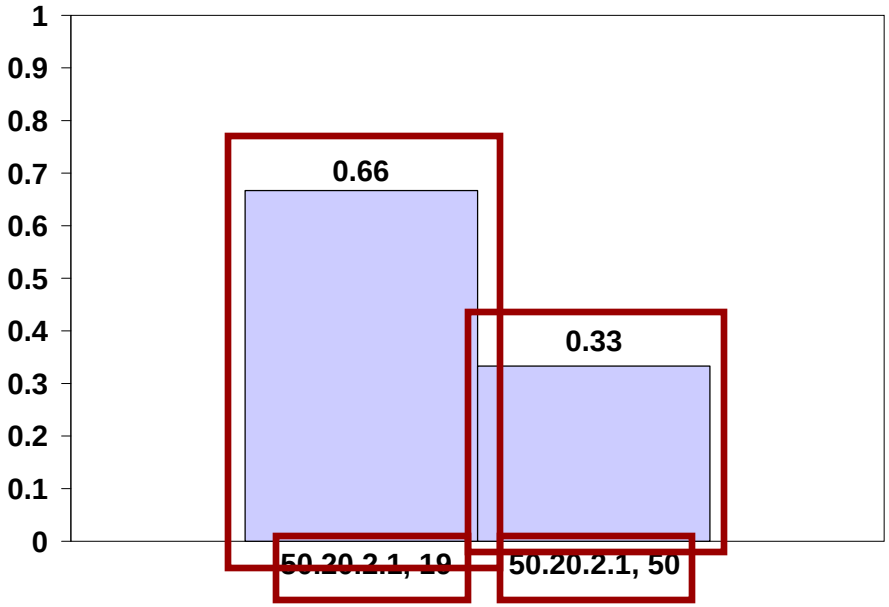
Conclusion

- Privacy risks are due to information encoded within the anonymized network data
- Provide one of the first methods for evaluating anonymized data for information leakage
  - Discover objects at risk of deanonymization
  - Compare anonymization policies and techniques
  - Simulate hypothetical attack scenarios

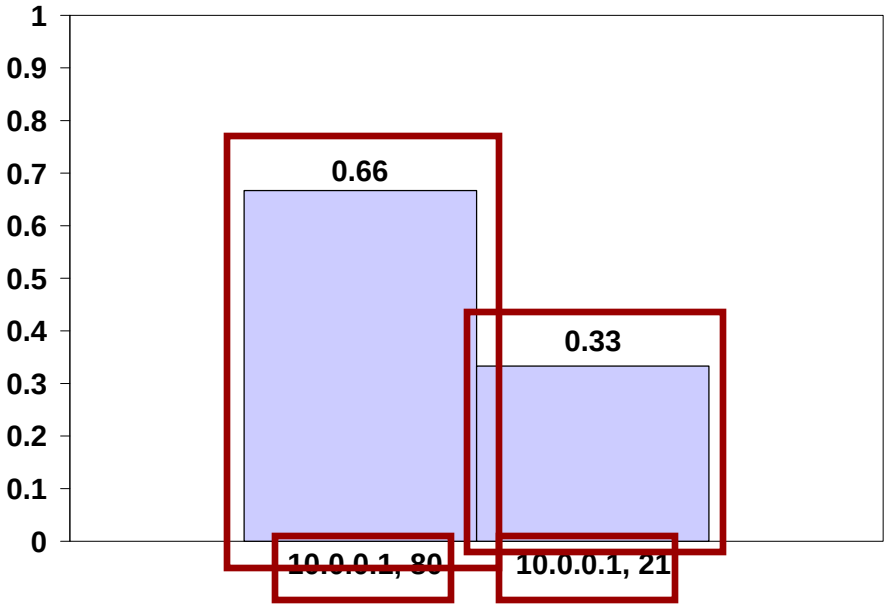
Calculating Anonymity

Local IP, Local Port Feature

Anonymized Object: 50.20.2.1



Unanonymized Object: 10.0.0.1

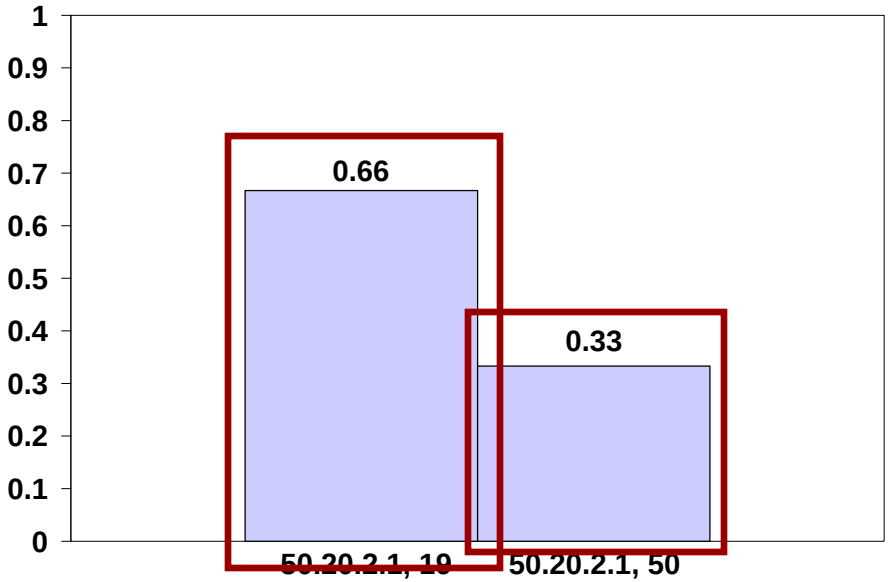


Unanon. Object	10.0.0.1
Similarity	2.0

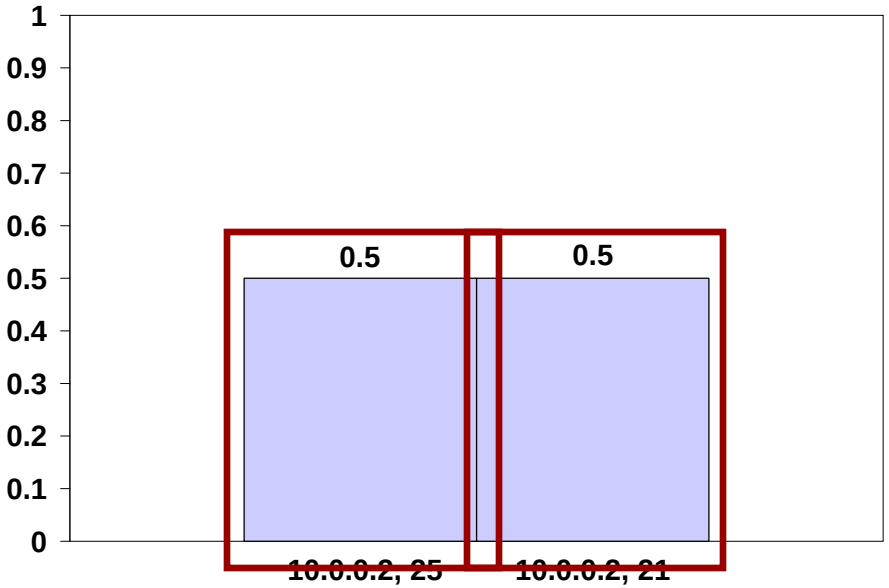
Calculating Anonymity

Local IP, Local Port Feature

Anonymized Object: 50.20.2.1



Unanonymized Object: 10.0.0.2

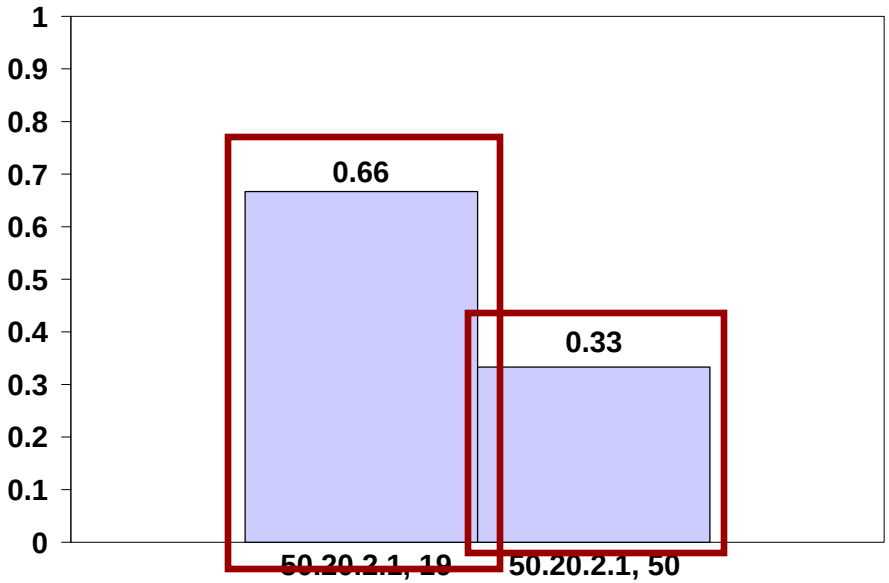


Unanon. Object	10.0.0.1	10.0.0.2
Similarity	2.0	1.66

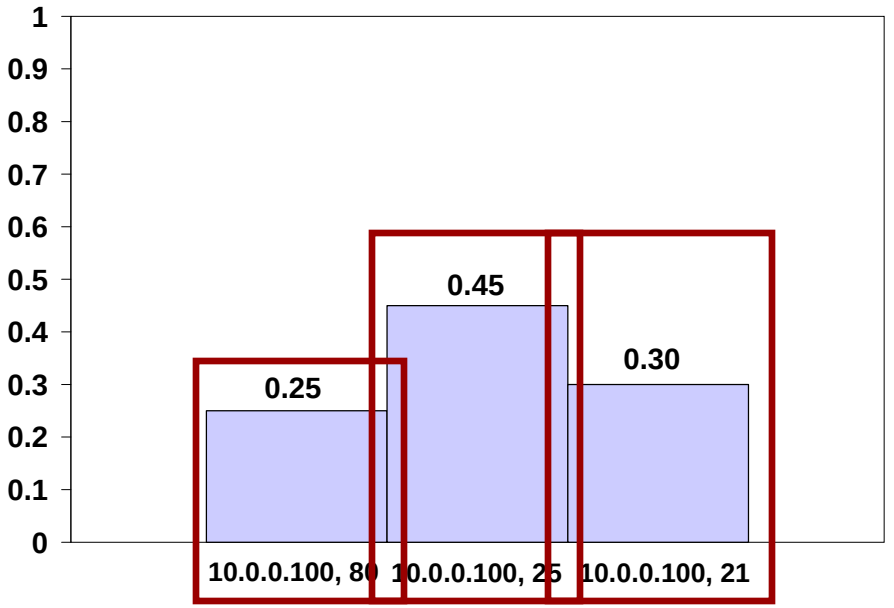
Calculating Anonymity

Local IP, Local Port Feature

Anonymized Object: 50.20.2.1



Unanonymized Object: 10.0.0.100



Unanon. Object	10.0.0.1	10.0.0.2	10.0.0.100
Similarity	38.70%	31.66%	29.21%