# CPS 590.5 Computer Security
# Lecture 13: Proof of Work

Xiaowei Yang
xwy@cs.duke.edu

# Roadmap

- Previous lecture
  - Capability-based DDoS defense
  - Filter-based DDoS defense

- Today
  - Proof of Work

# Observation

- DDoS defenses at best offer fairness
  - Can't distinguish good from bad

- Different ways of providing fairness
  - TVA
  - StopIt
  - SpeakUp
  - Portcullis

# Portcullis: Protecting Connection Setup from Denial-of-Capability Attacks

Bryan Parno, Dan Wendlandt, Elaine Shi, Adrian Perrig, Bruce Maggs, Yih-Chun Hu

# DDOS

- Distributed DoS attack exhausts bandwidth of links leading to victim.

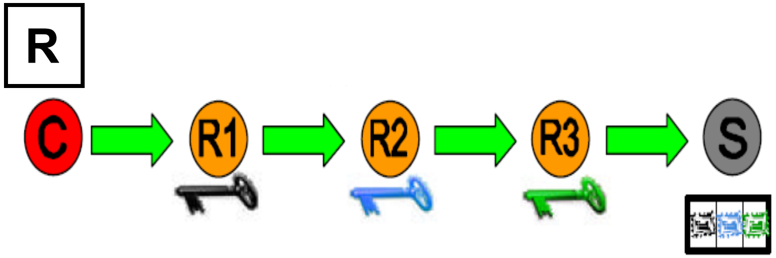
- The victim of a network DDoS attack can often **identify** legitimate traffic flows but lacks the ability to give these flows prioritized access to the bottleneck link.

- Routers have the power to **prioritize** traffic, but cannot effectively identify legitimate packets without input from the receiver.

# Capability System

capability-based systems treat prioritized traffic preferentially over non-prioritized traffic.

- Network capabilities enable a receiver to inform routers of its desire to prioritize particular flows.
- To set up a Network Capability
  1. The source sends a capability request packet to the destination.
  2. Routers, on the path, add cryptographic markings to the packet header.
  3. The receiver accumulates routers' markings to represent the capability.
  4. The receiver then permits a flow by returning the capability to the sender.
  5. Sender includes the capability in subsequent packets to receive prioritized service from the routers.

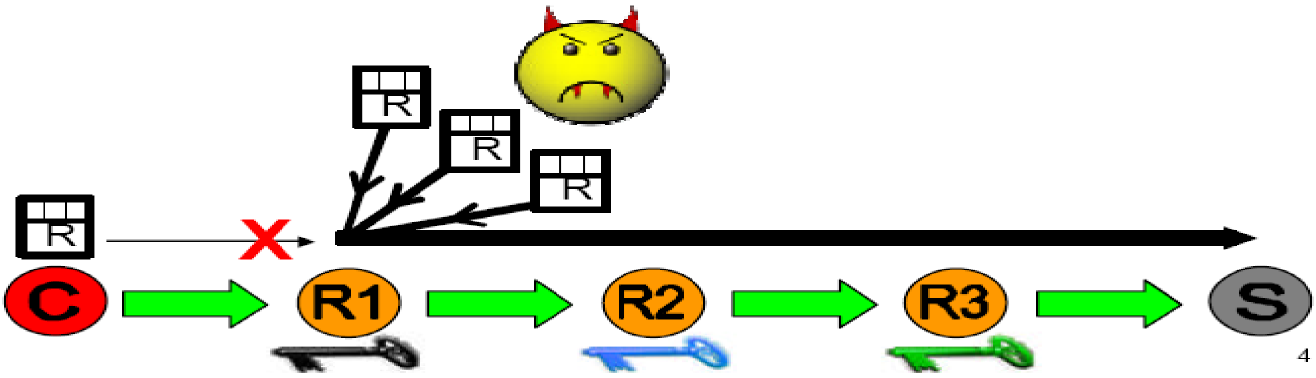


# DOC Attack

- An attacker can flood the capability-setup channel…

  …DOC Attack!!!

- To prevent DoC Attack…Capability systems need a DDoS defense mechanism.



# DOC Defenses

## Identity based Fairness

- Per-Source Fairness $r'_g = min\ (r_g, \gamma / (n_g+m))$
  - An adversary can easily spoof its IP address, and sources behind large NATs may be subject to grossly unfair treatment.

- Per-Path Fairness $r'_g = min\ (r_g, \gamma / (|P| * N_{pi}))$
  - Attackers are still able to spoof paths by inserting bogus initial markings in the path ID field.
  - This increase |P| and creates small values of $N_{pi}$

- Per-Destination Fairness
  - An attacker can flood packets to all destinations that share the victim's bottleneck link.

# DOC Defenses (Cont'd)

## Proof-of-Work Schemes

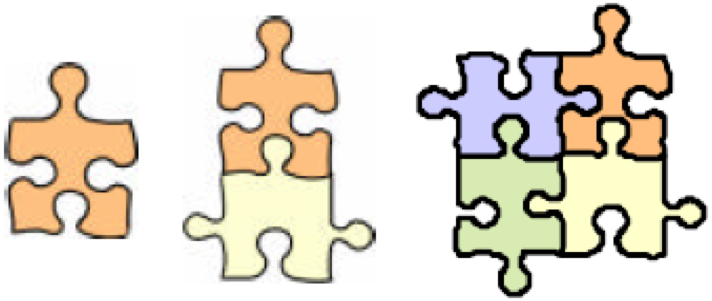
- Per-Bandwidth Fairness $r'_g = min\ (r_g, \gamma * \kappa / K)$
  - Hosts sending to destinations other than the victim may experience congestion because of the increase in traffic from legitimate senders.
  - Large disparities can exist in the amount of bandwidth available to legitimate users.

- Per-Computation Fairness $r'_g = min\ (r_g, \gamma * c_g / C)$

*Portcullis is based on Per-Computation Fairness*

# Portcullis

- **Goal**
  - Portcullis aims to provide a strong defense against large-scale DDoS attacks.
  - Portcullis designs a DoC-resistant request channel for a capability system.
  - Portcullis design is based on computational puzzles.

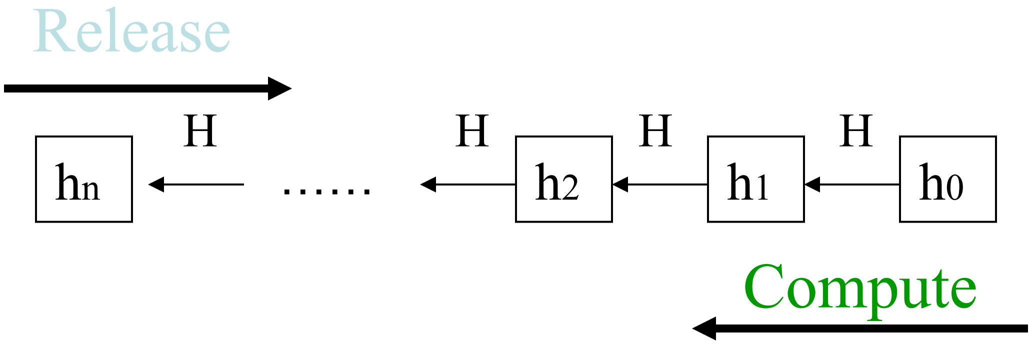


# Portcullis (Cont'd)

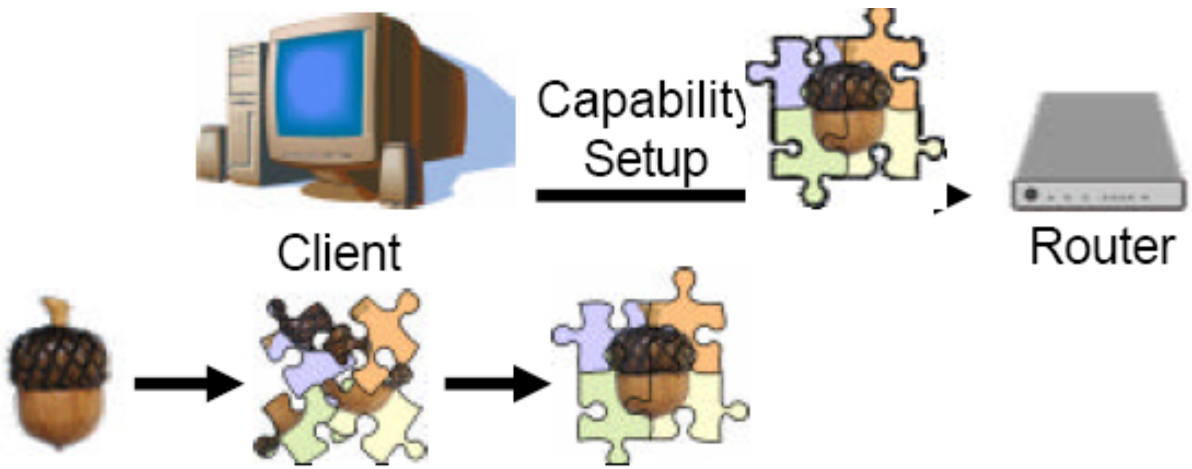
1. Sender obtains the latest seed from the seed distribution service.
2. Sender generates a puzzle using the puzzle generation algorithm.
3. The sender then computes the solution to the puzzle.
4. Sender includes the puzzle and solution in the header of the request packet.
5. The Routers verify the authenticity of the puzzle and the solution, and give priority to requests containing higher-level puzzles.

# 1. Seed Generation

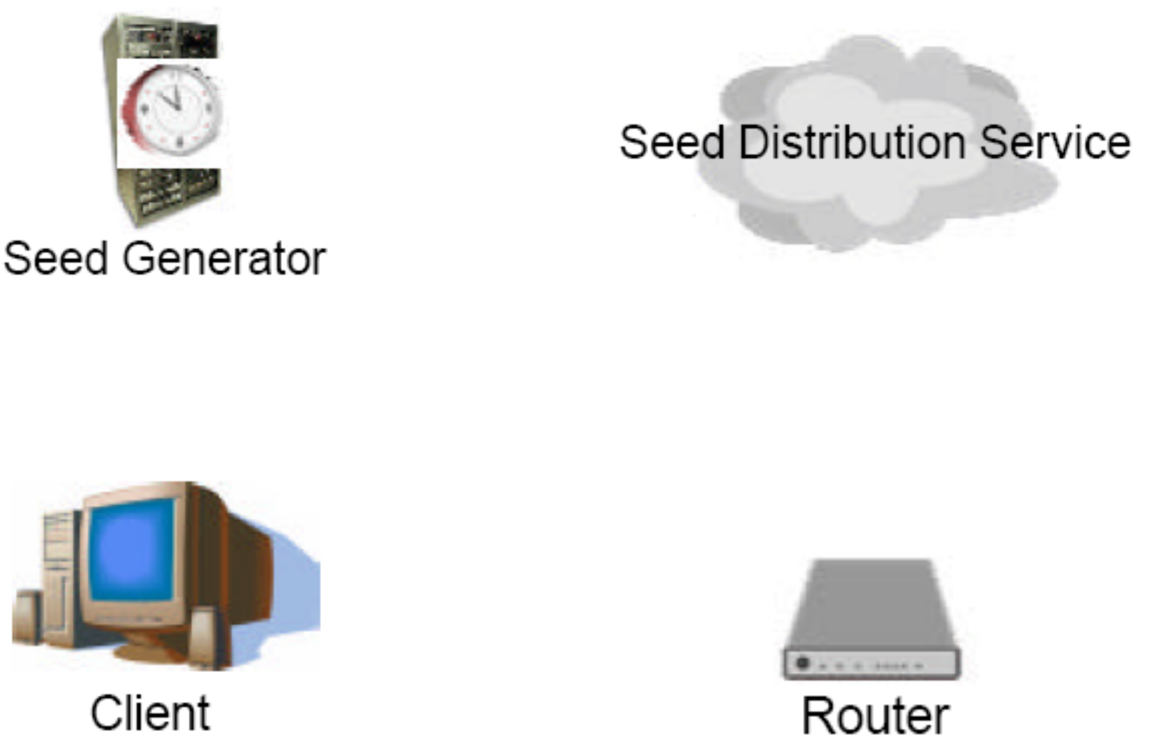
- The **seed generator** periodically releases a new seed for senders to use in creating puzzles.
- Puzzle seed must be:
  - Unpredictable.
  - Efficiently verifiable.



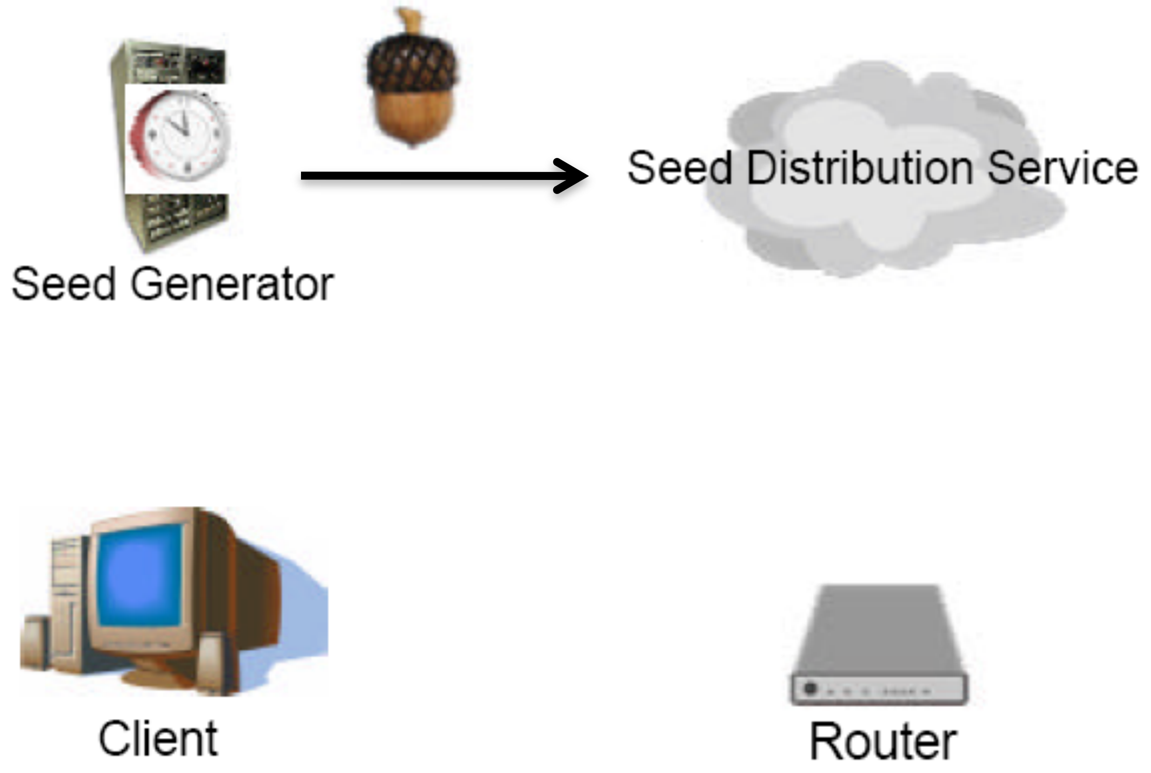
# 2. Seed Distribution



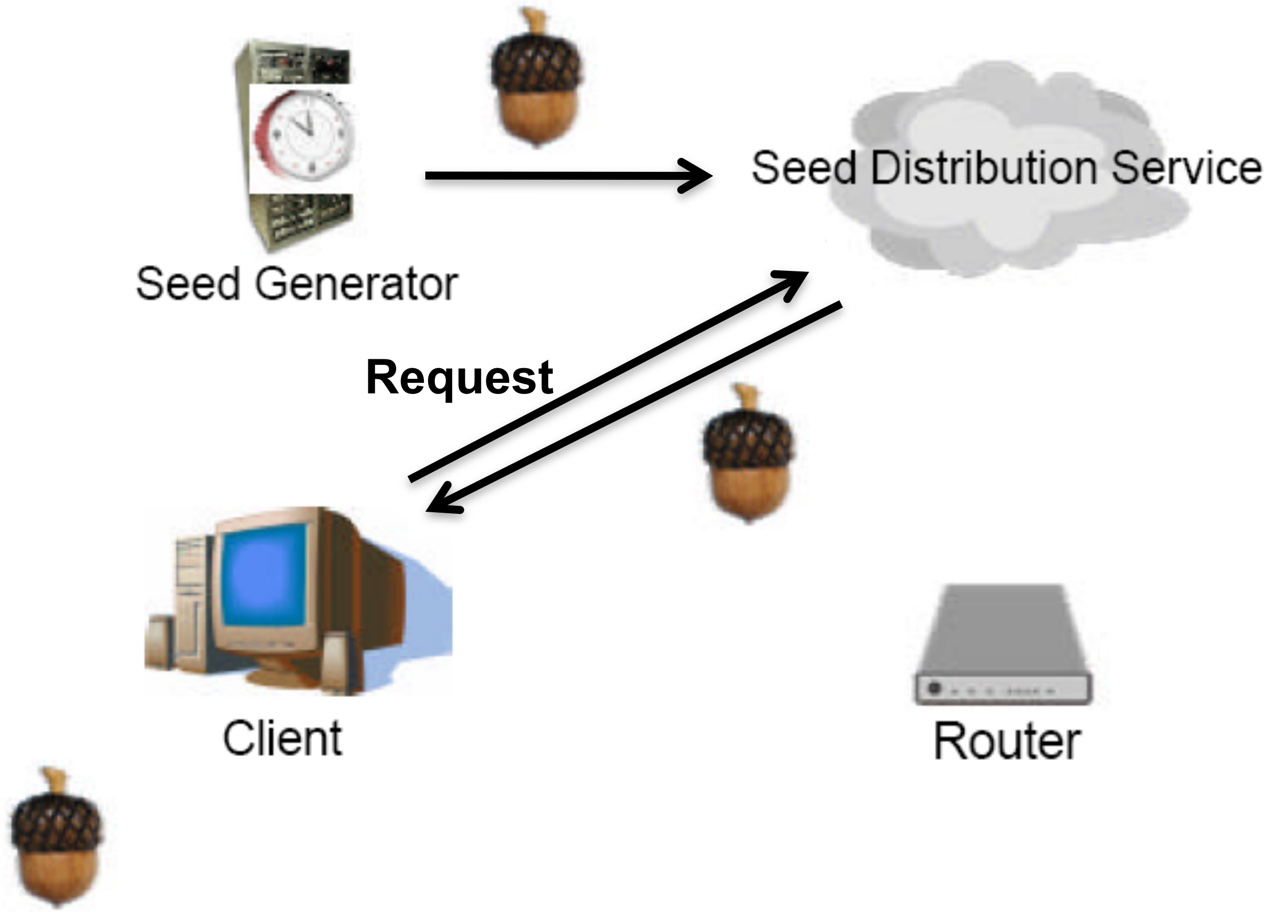
# 2. Seed Distribution



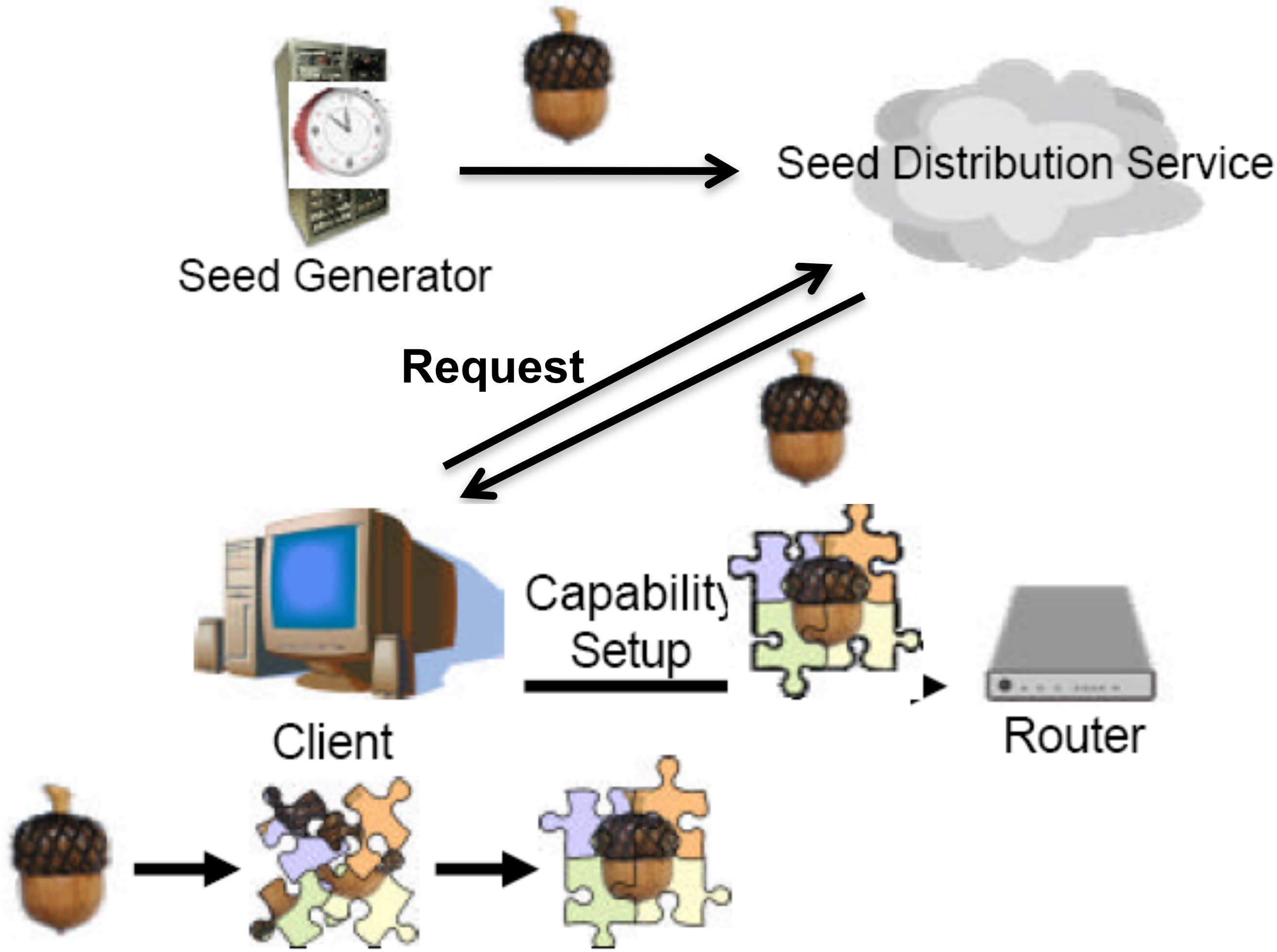
# 2. Seed Distribution



# 2. Seed Distribution



# 2. Seed Distribution



# 3. Puzzle Generation

- Client computes a flow-specific puzzle as:

$$P \longleftarrow H(\text{ Server IP } \| S \| R \| L \| X)$$

- Where:
  - ✓ H is a hash function
  - ✓ S is the current puzzle seed
  - ✓ R is a randomly chosen 64-bit number
  - ✓ L is the puzzle difficulty level

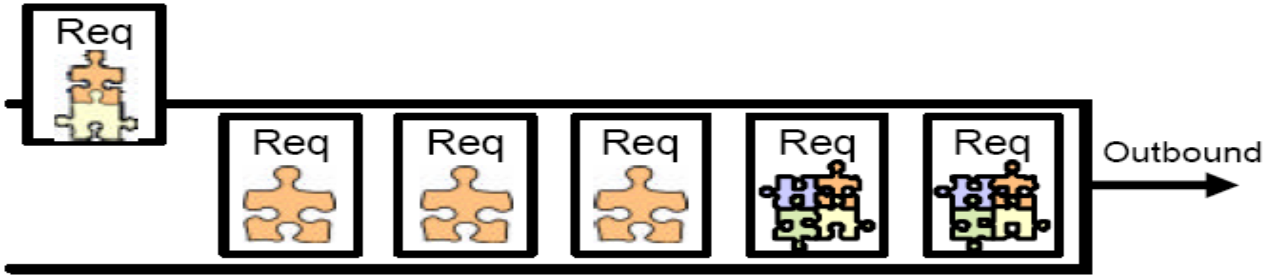
- The solution **X** is chosen so that the **last L bits of p are all zero.**

# 3. Router Verification and Scheduling

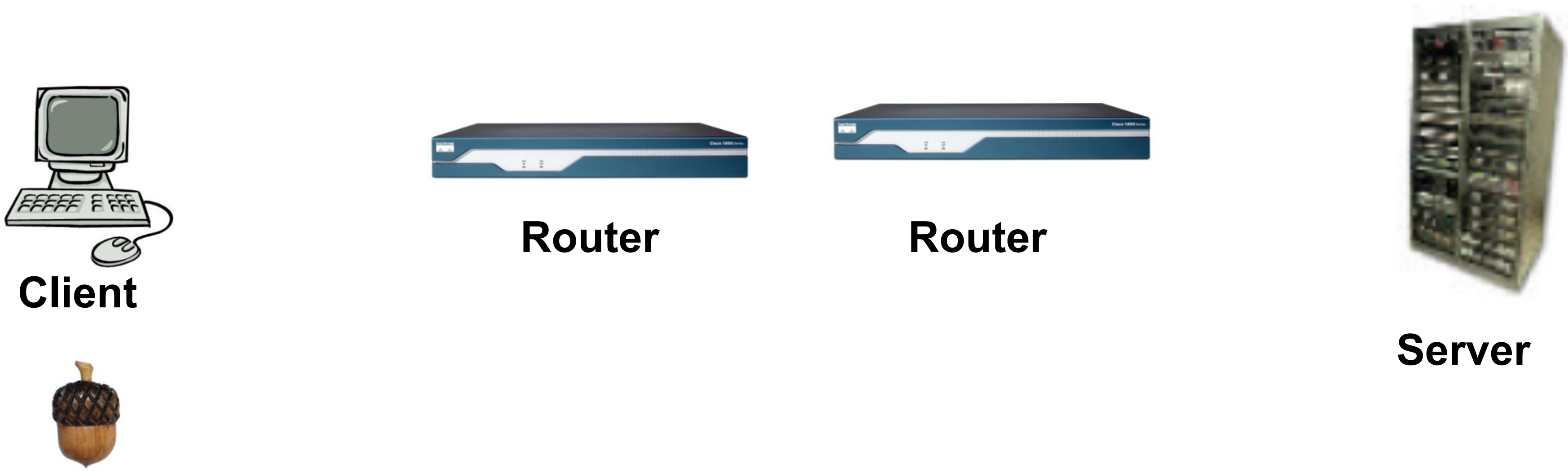
- The router **verify** the sender's puzzle solution by computing the same hash.

  $$P \longleftarrow H(\text{ Server IP } \| \text{ S } \| \text{ R } \| \text{ L } \| \text{ X})$$

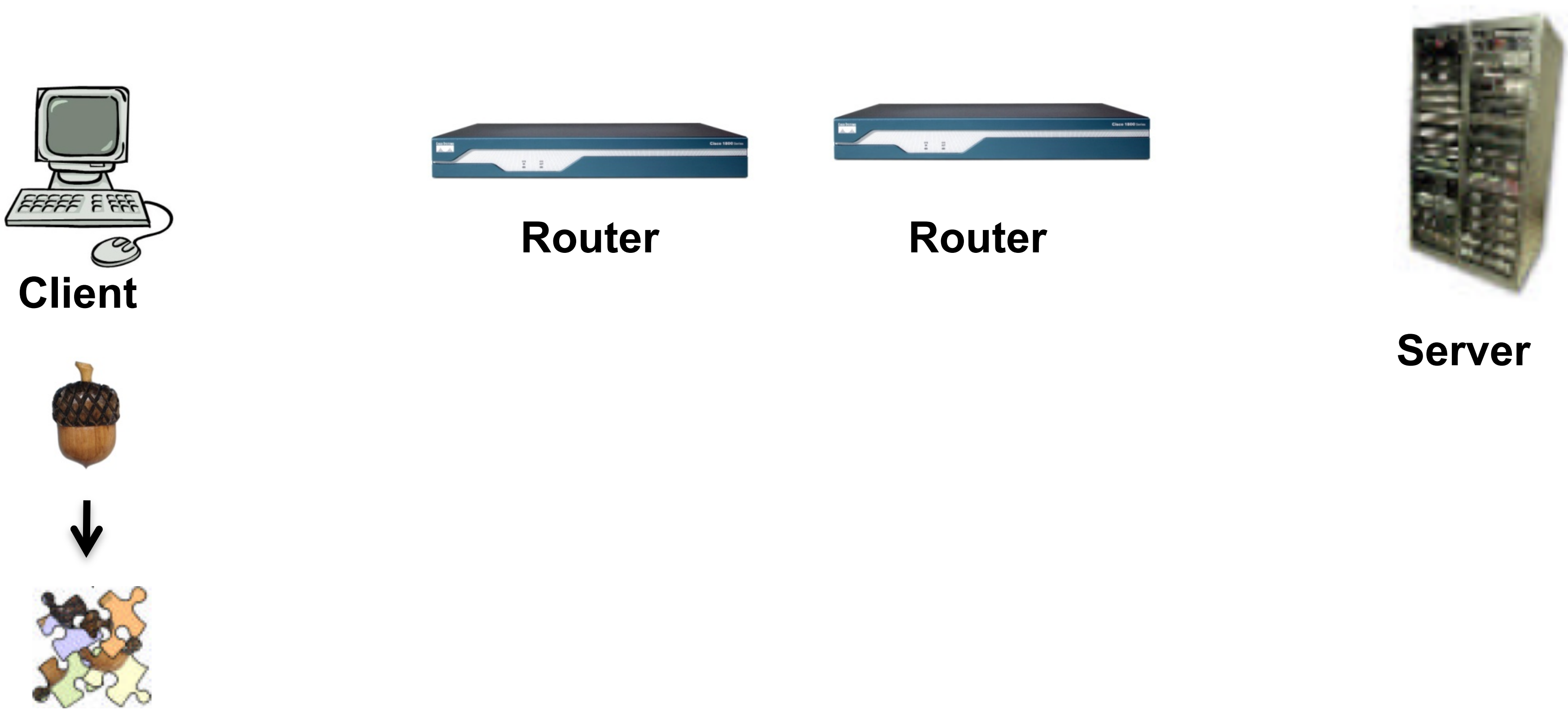
- Only correct puzzle solutions are entered into a Bloom filter.
  - Bloom Filter is configured to detect the reuse of puzzle solutions seen in the past period t.
- Prioritize packets with harder puzzles (larger L)



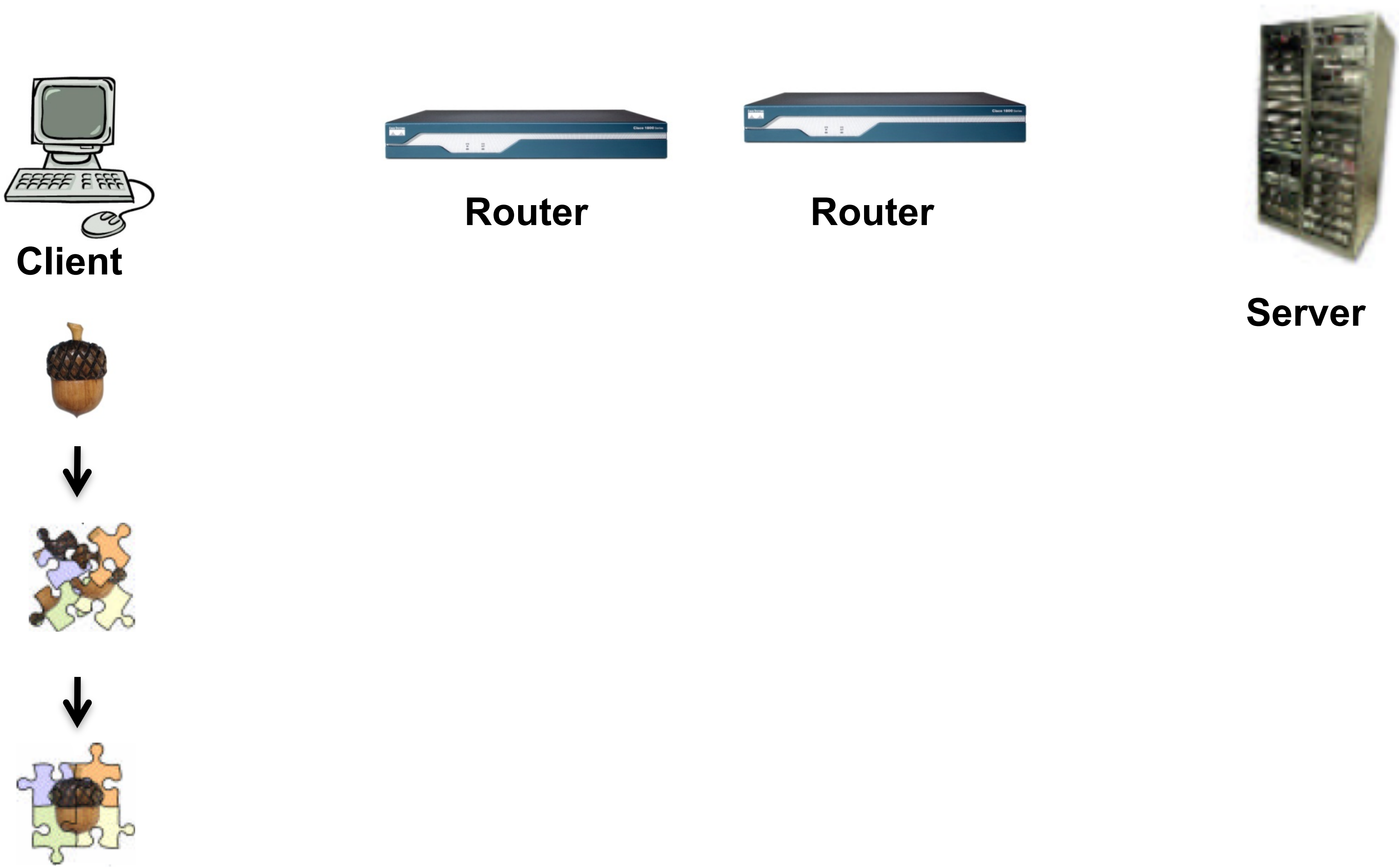
# Legitimate Sender Strategy



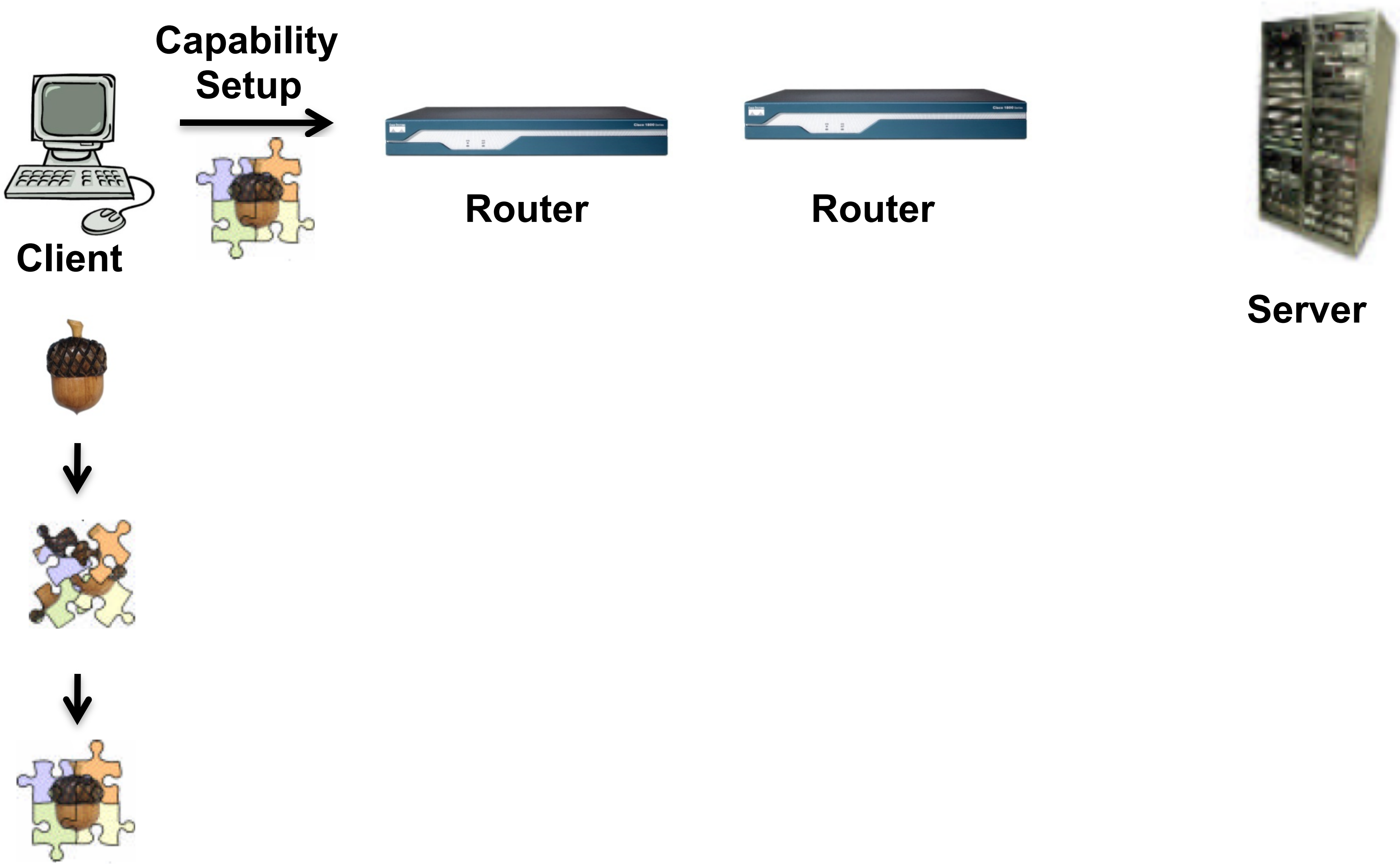
# Legitimate Sender Strategy



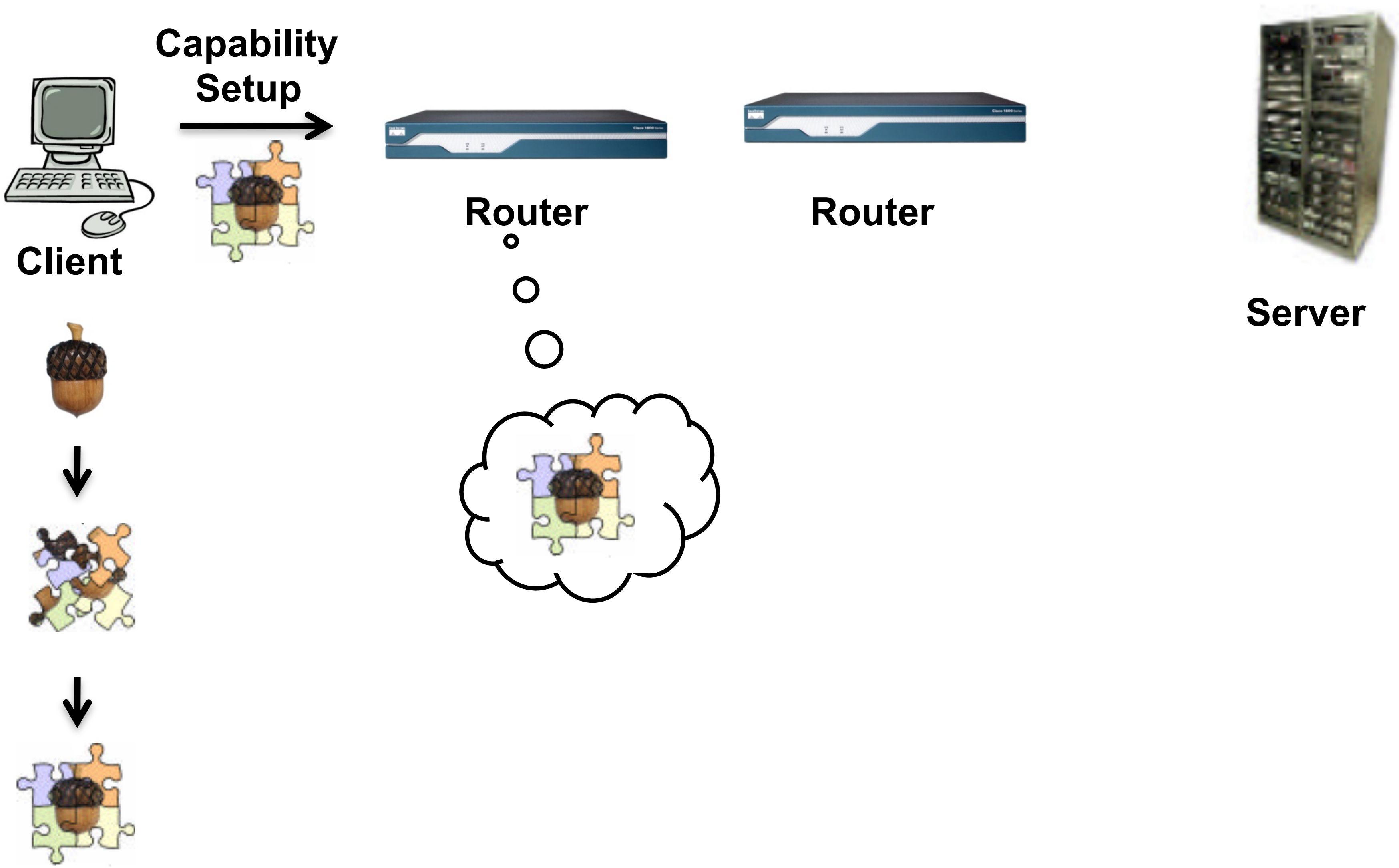
# Legitimate Sender Strategy



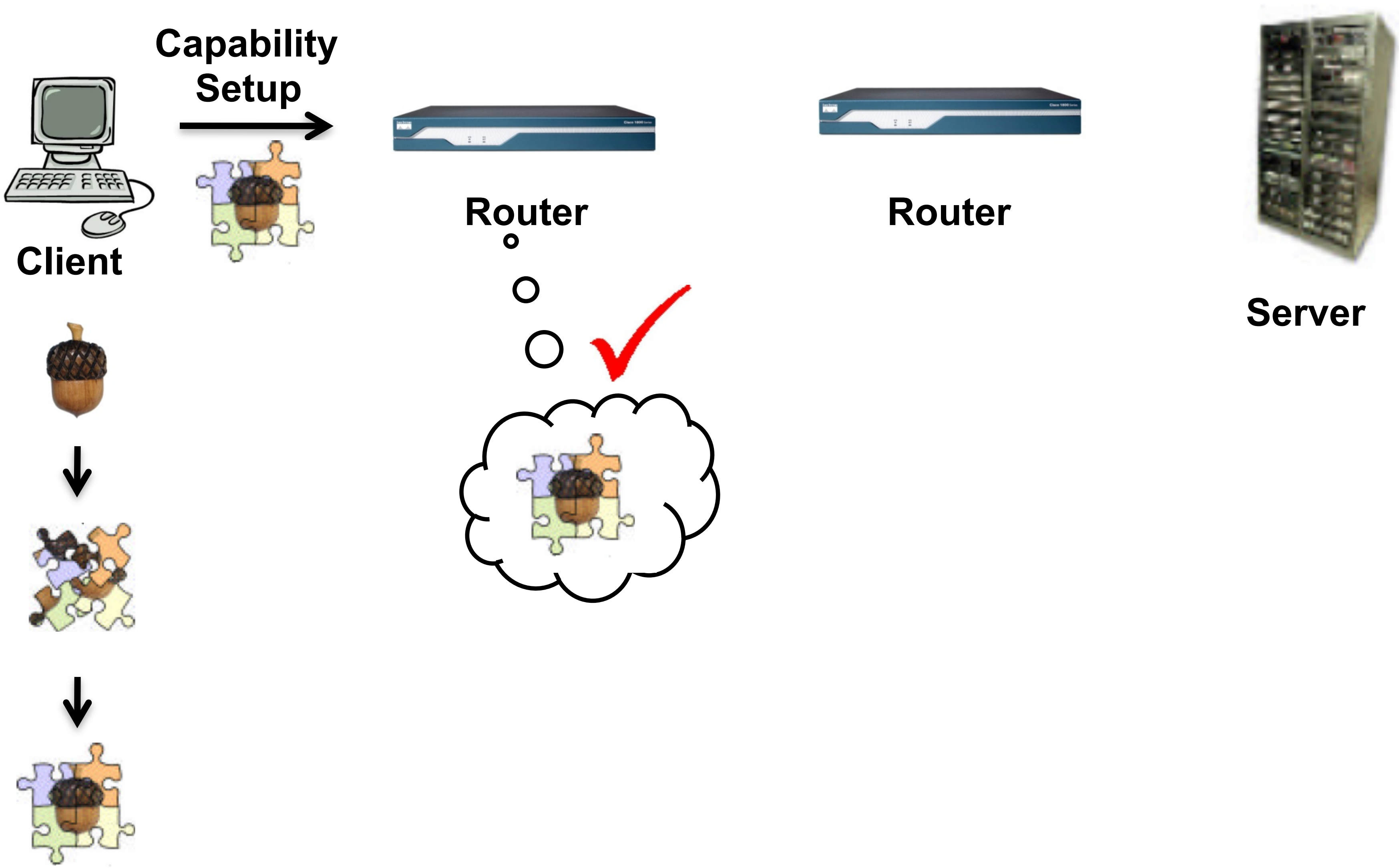
# Legitimate Sender Strategy



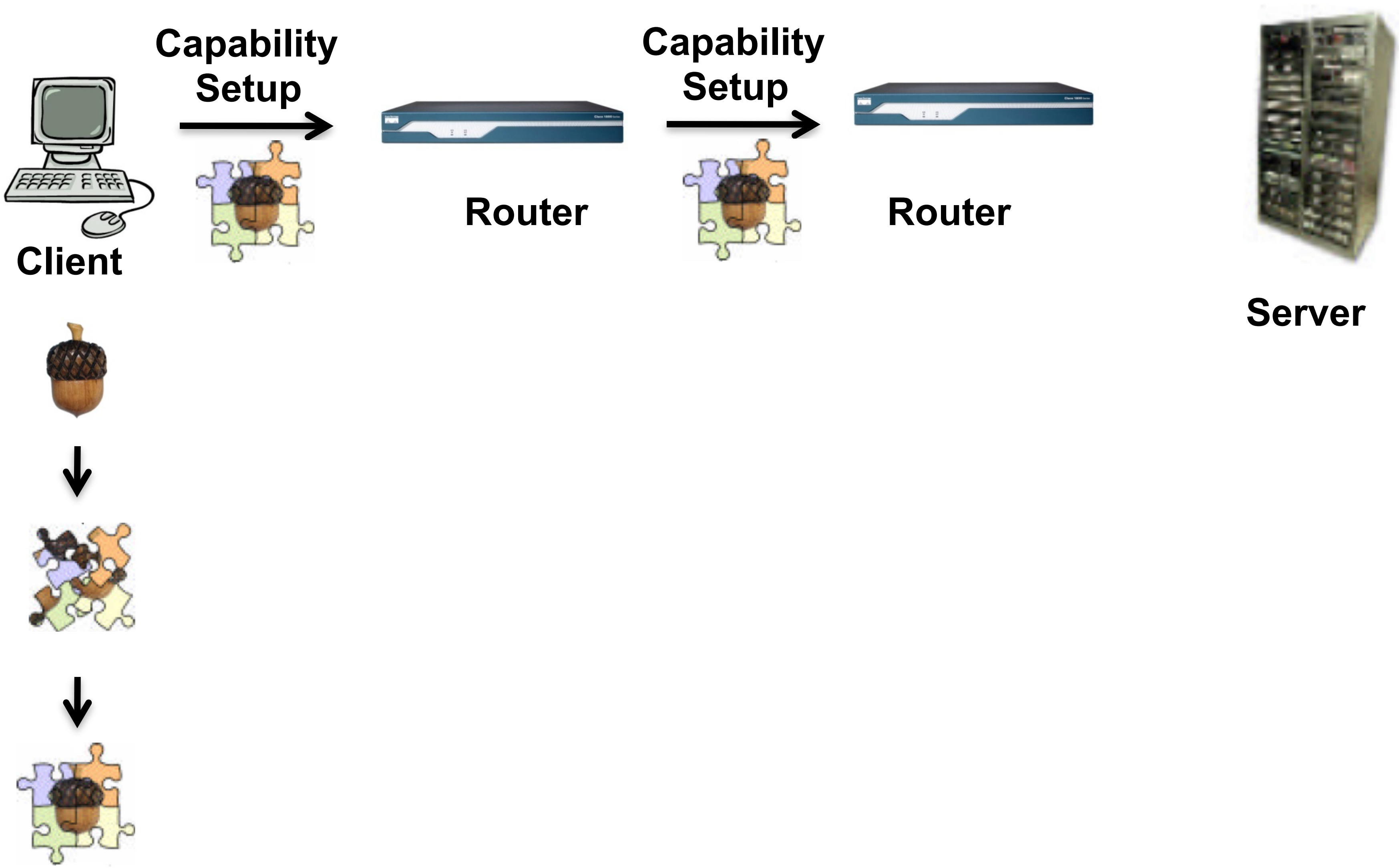
# Legitimate Sender Strategy



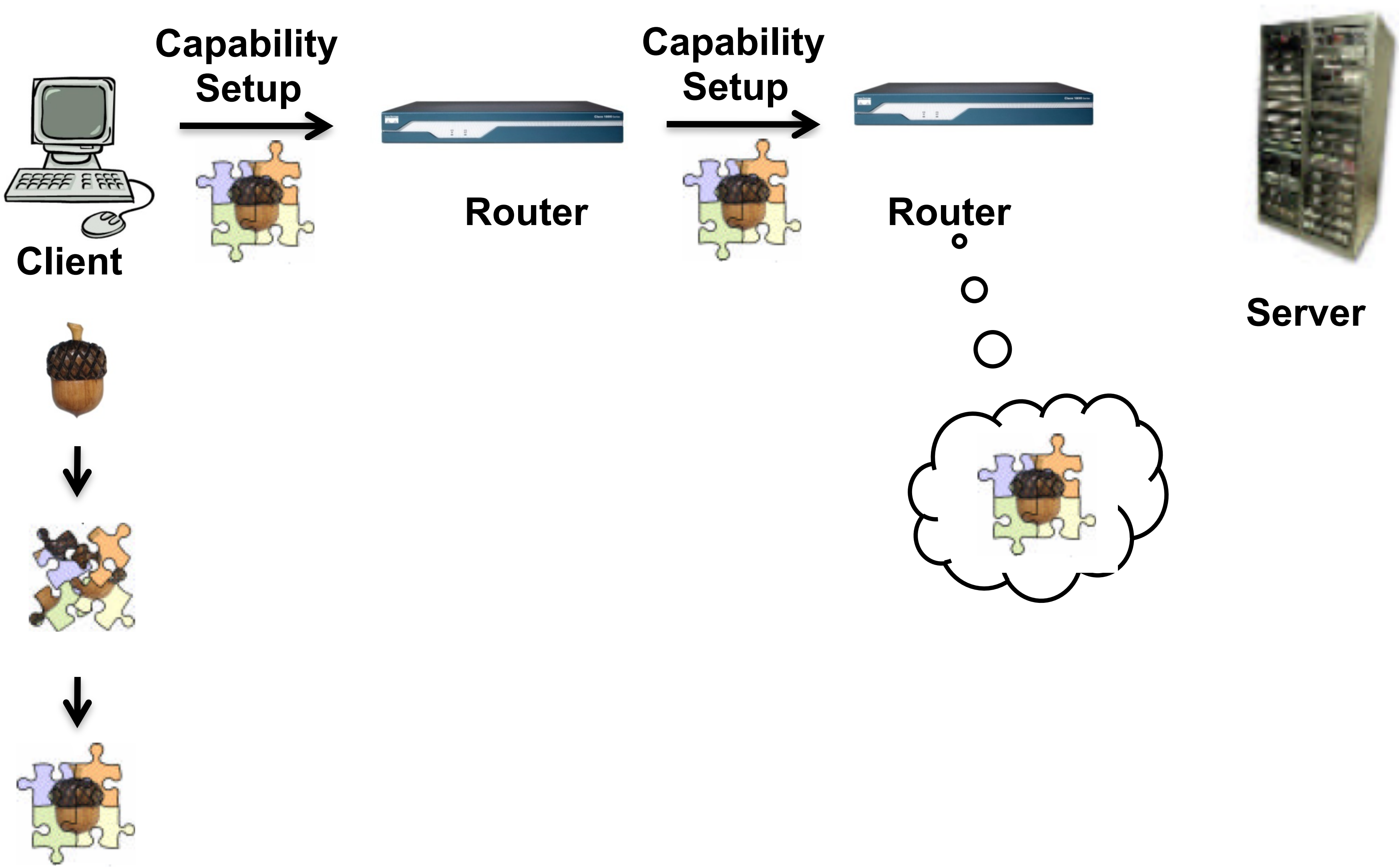
# Legitimate Sender Strategy



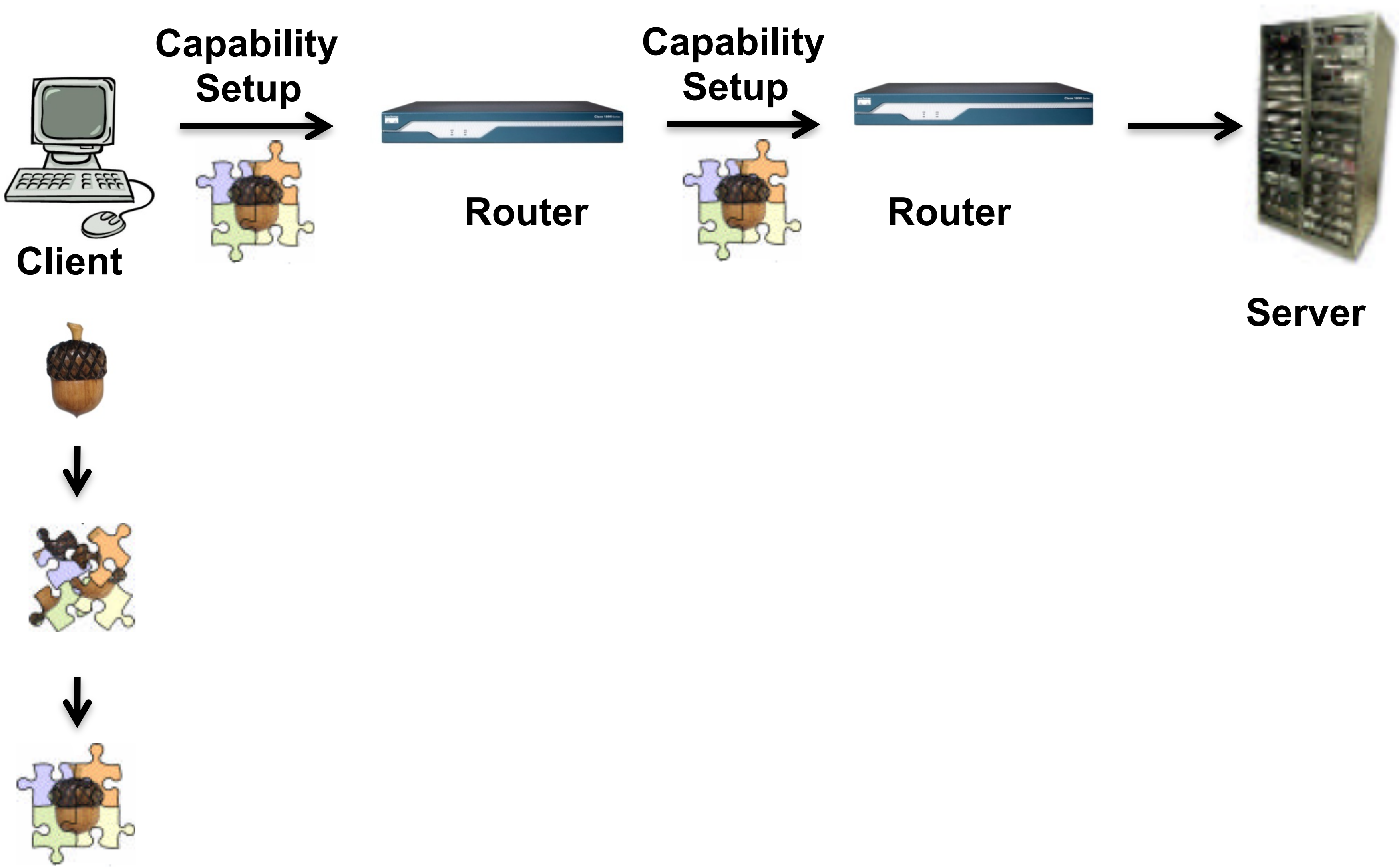
# Legitimate Sender Strategy



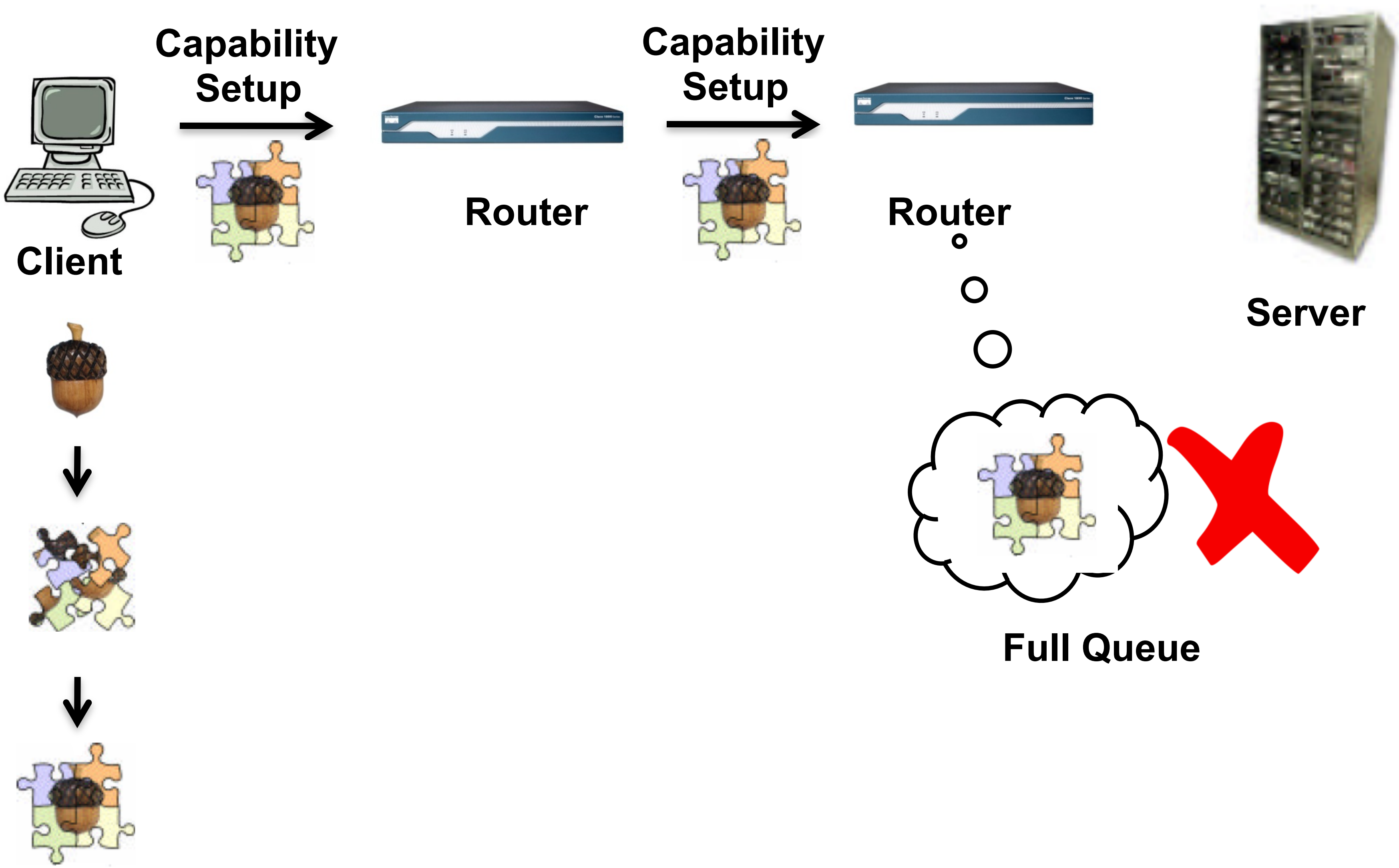
# Legitimate Sender Strategy



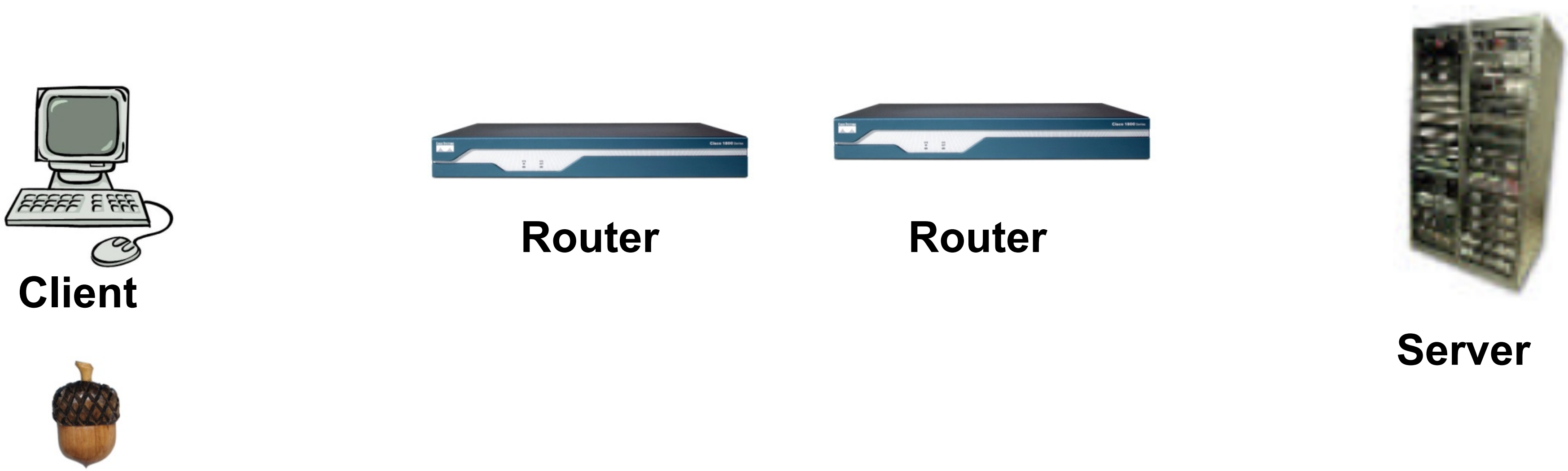
# Legitimate Sender Strategy



# Legitimate Sender Strategy



# Legitimate Sender Strategy



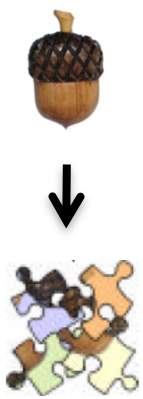
# Legitimate Sender Strategy

Client

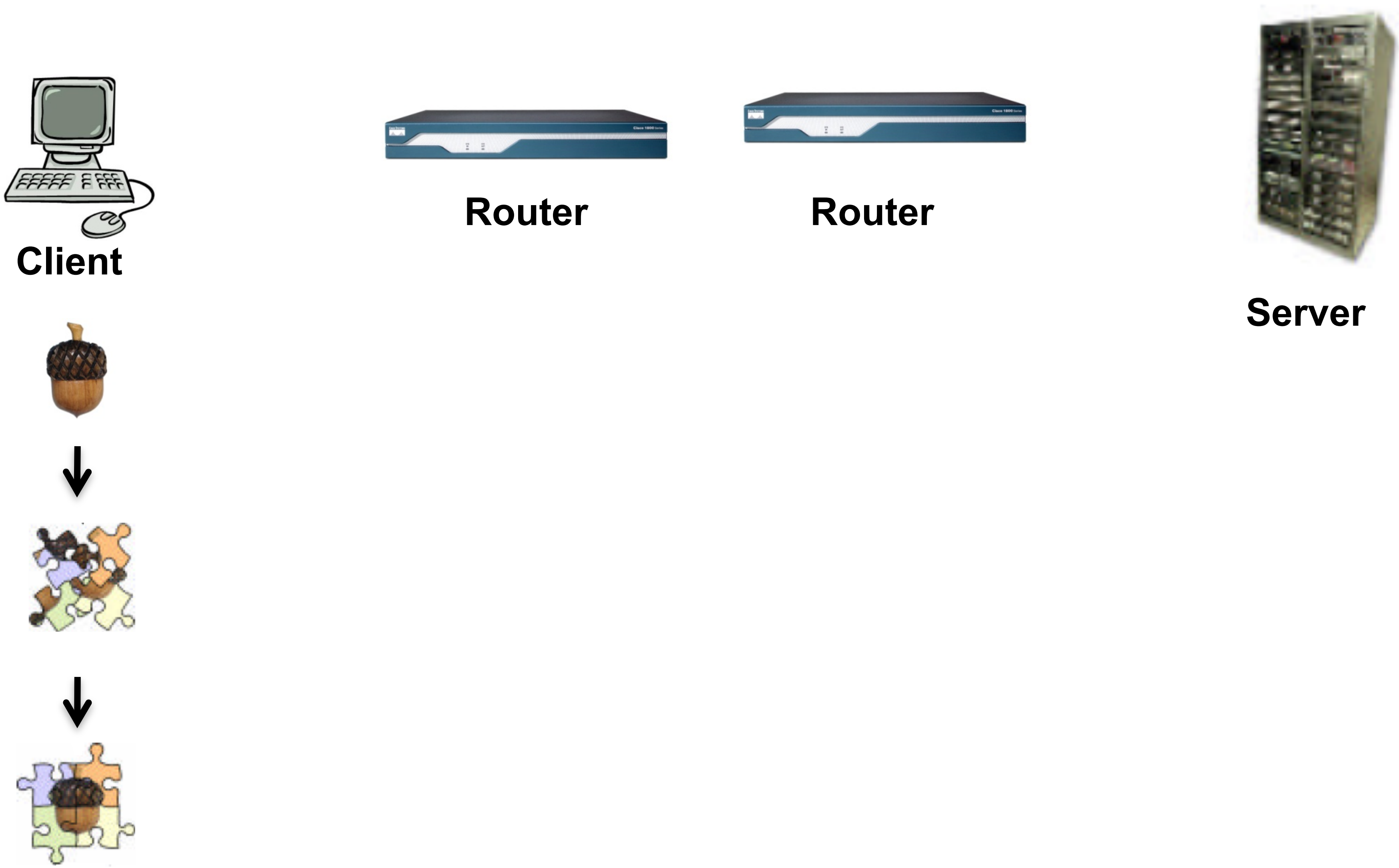
Router

Router

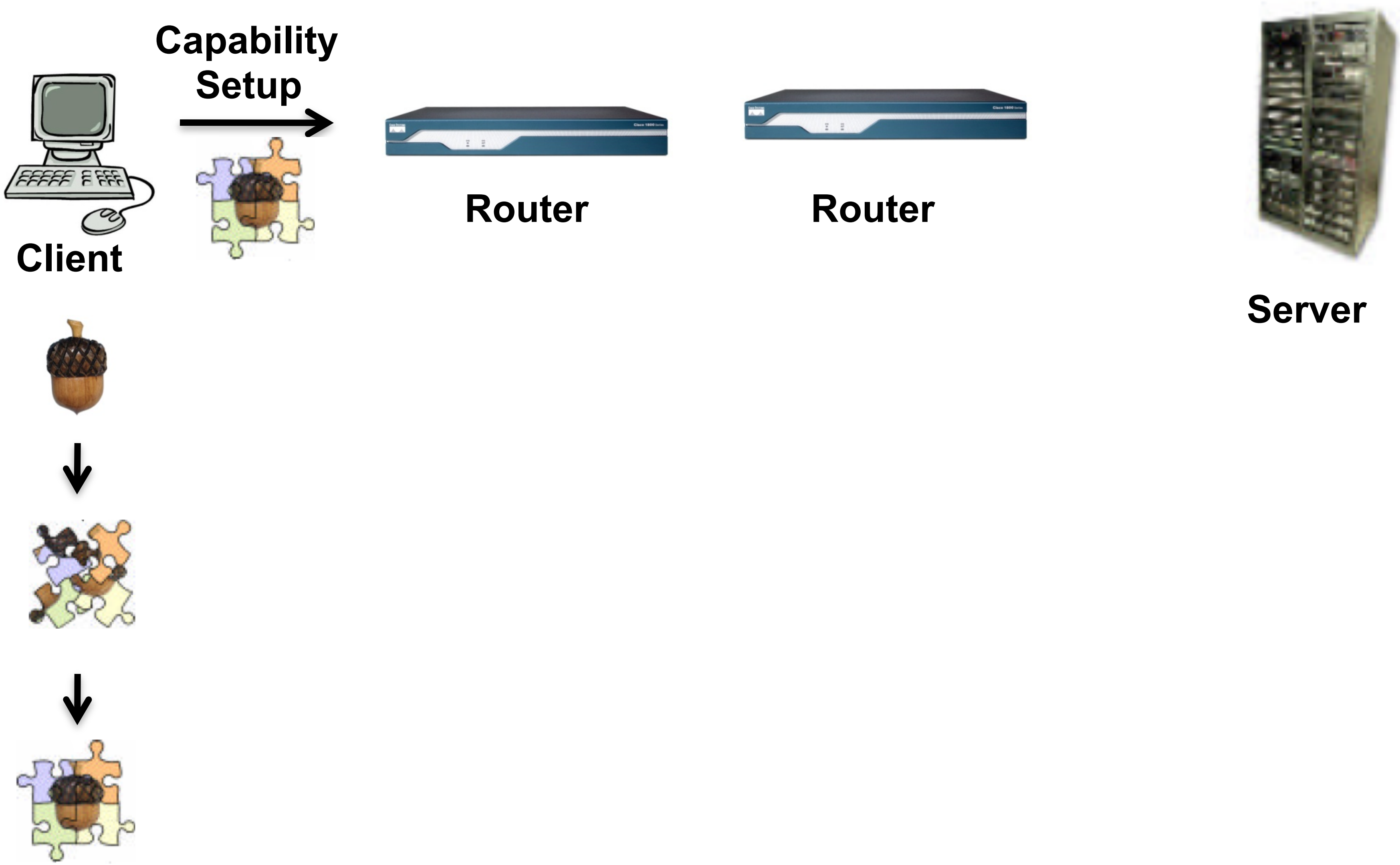
Server



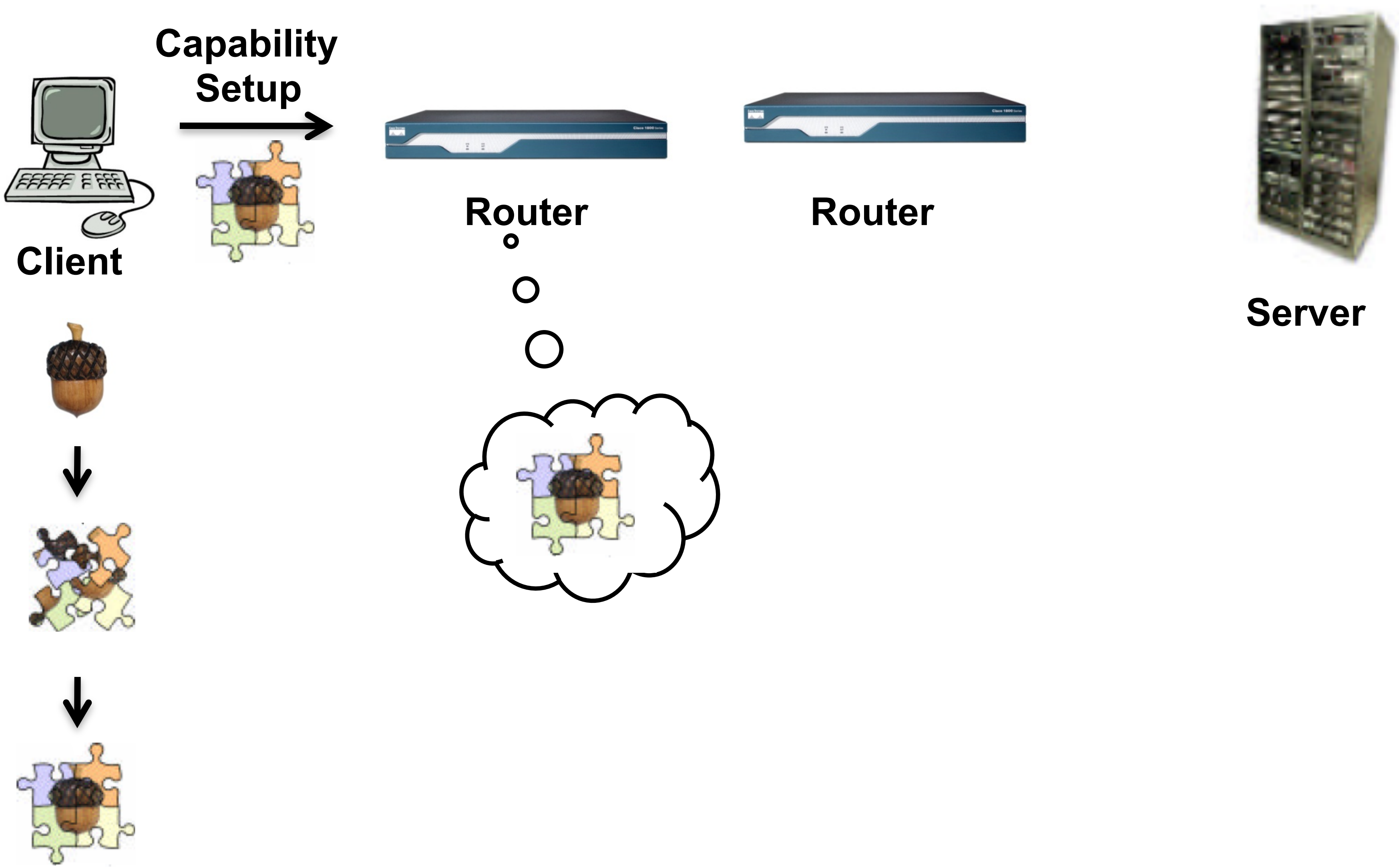
# Legitimate Sender Strategy



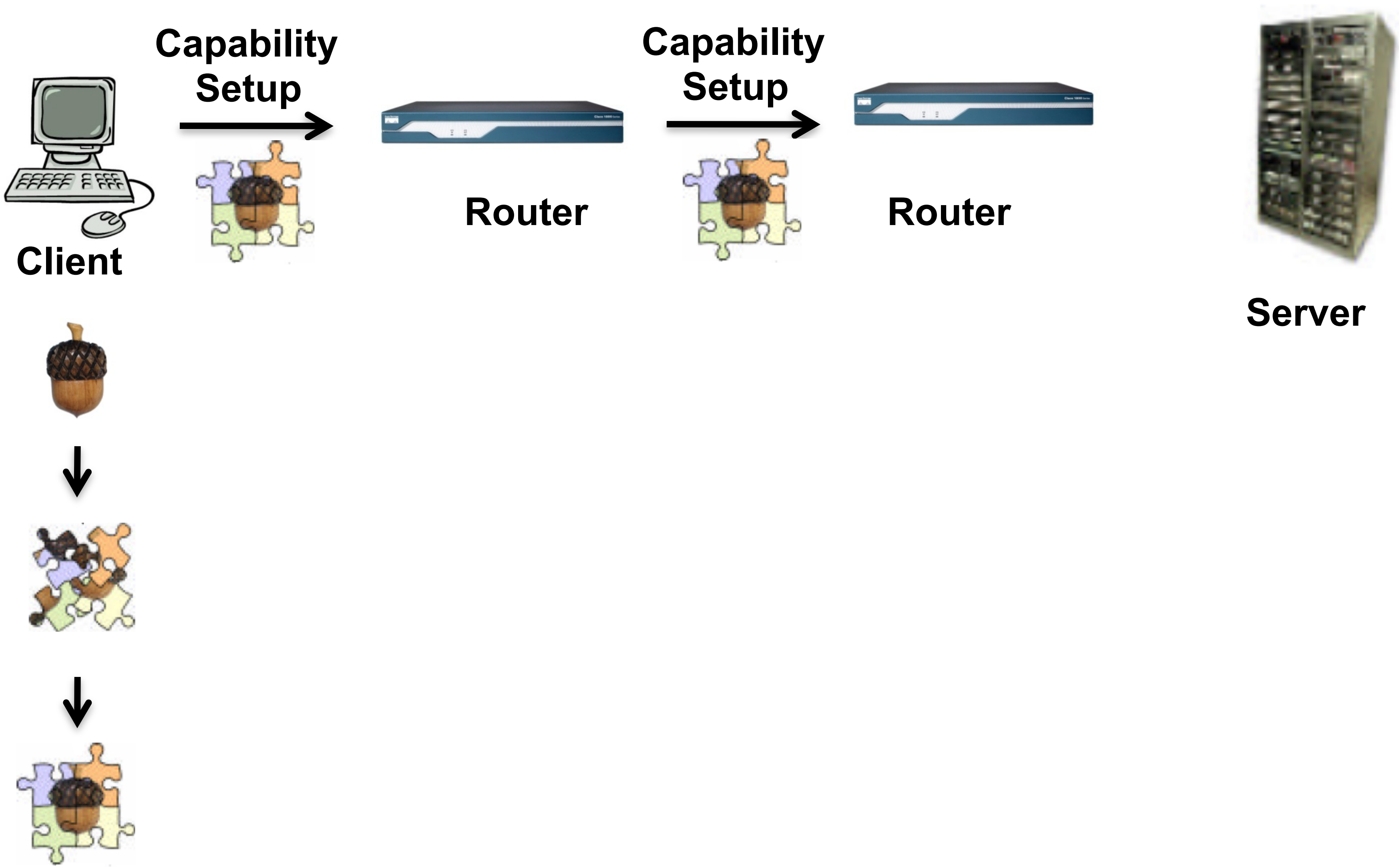
# Legitimate Sender Strategy



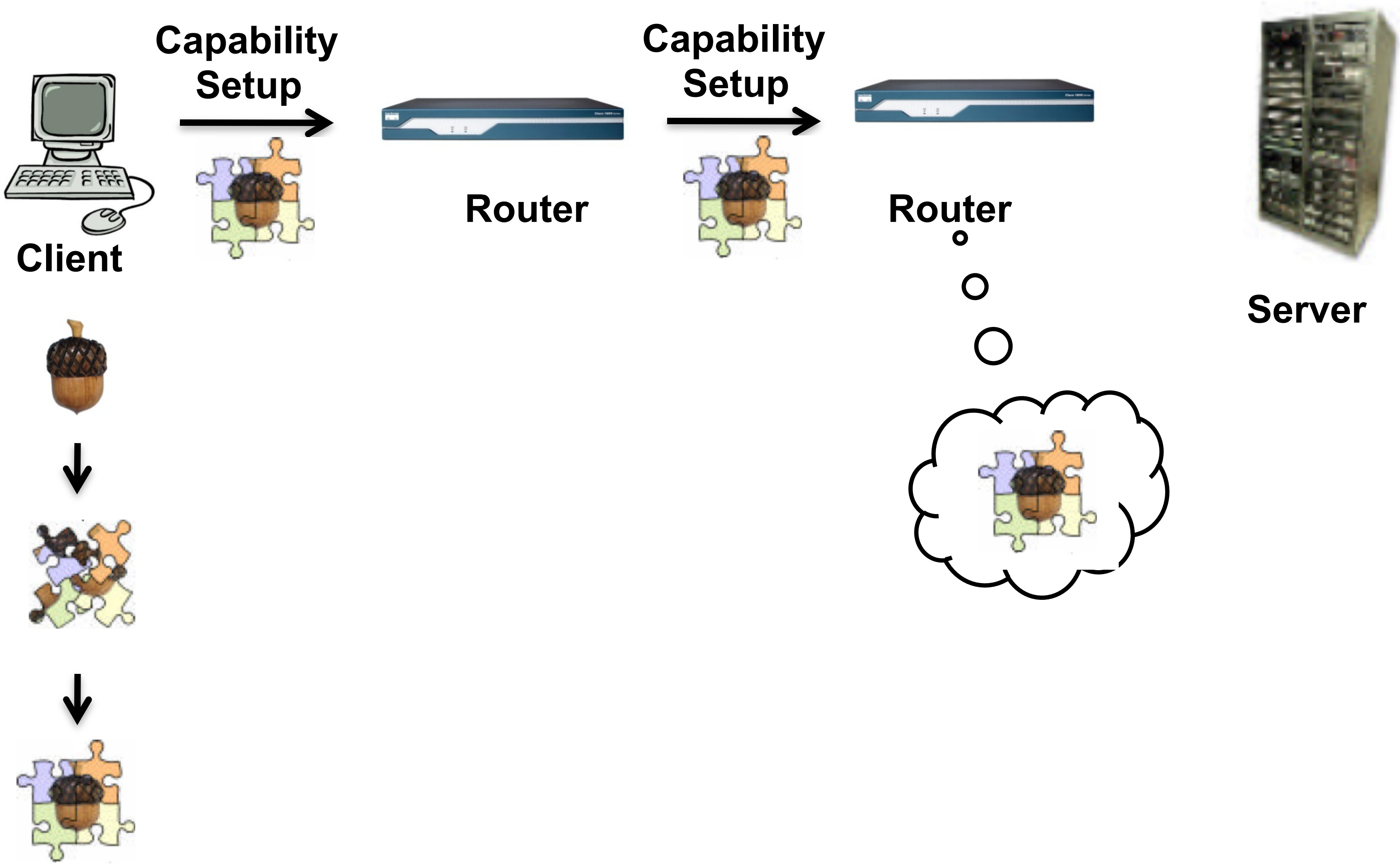
# Legitimate Sender Strategy



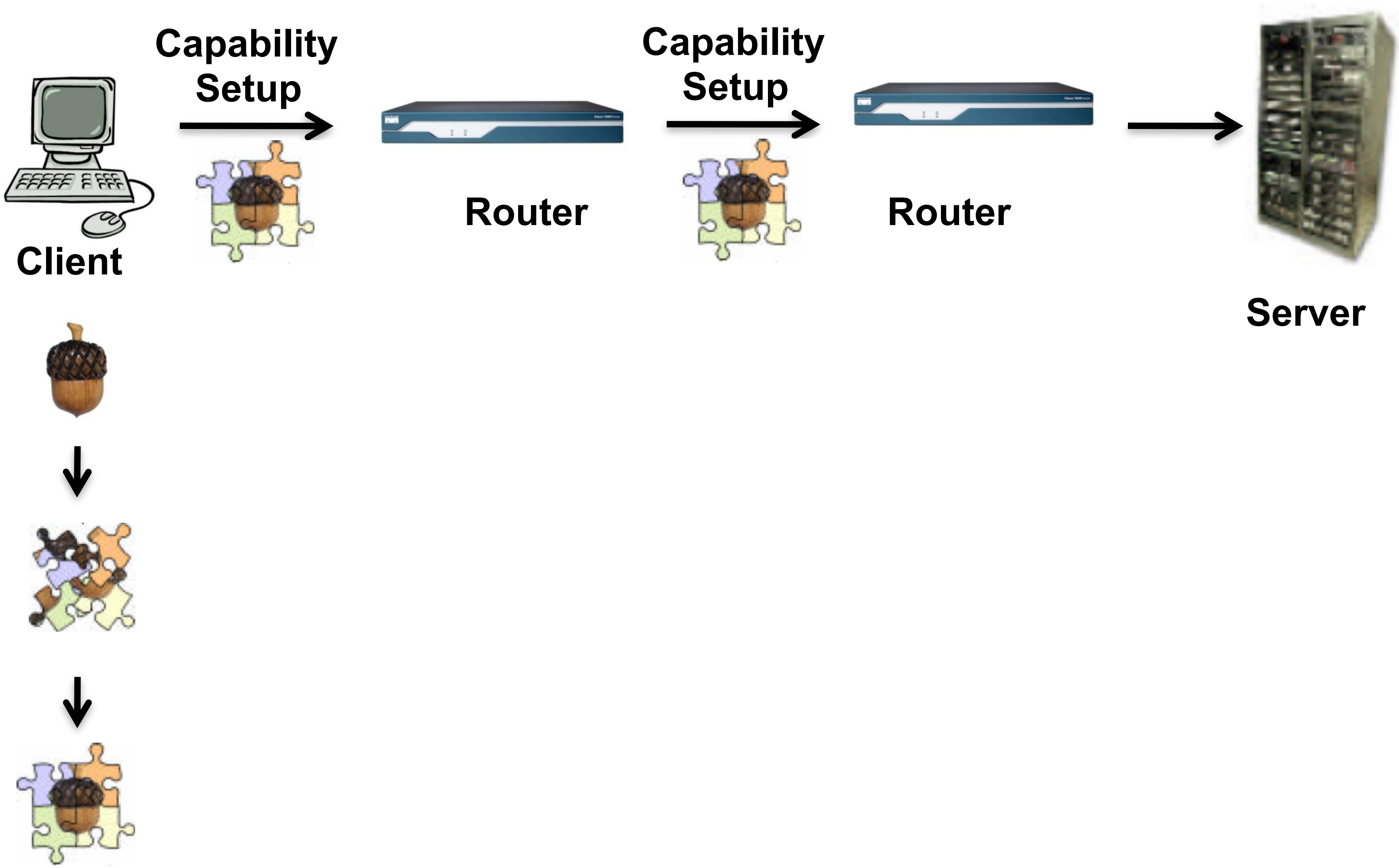
# Legitimate Sender Strategy



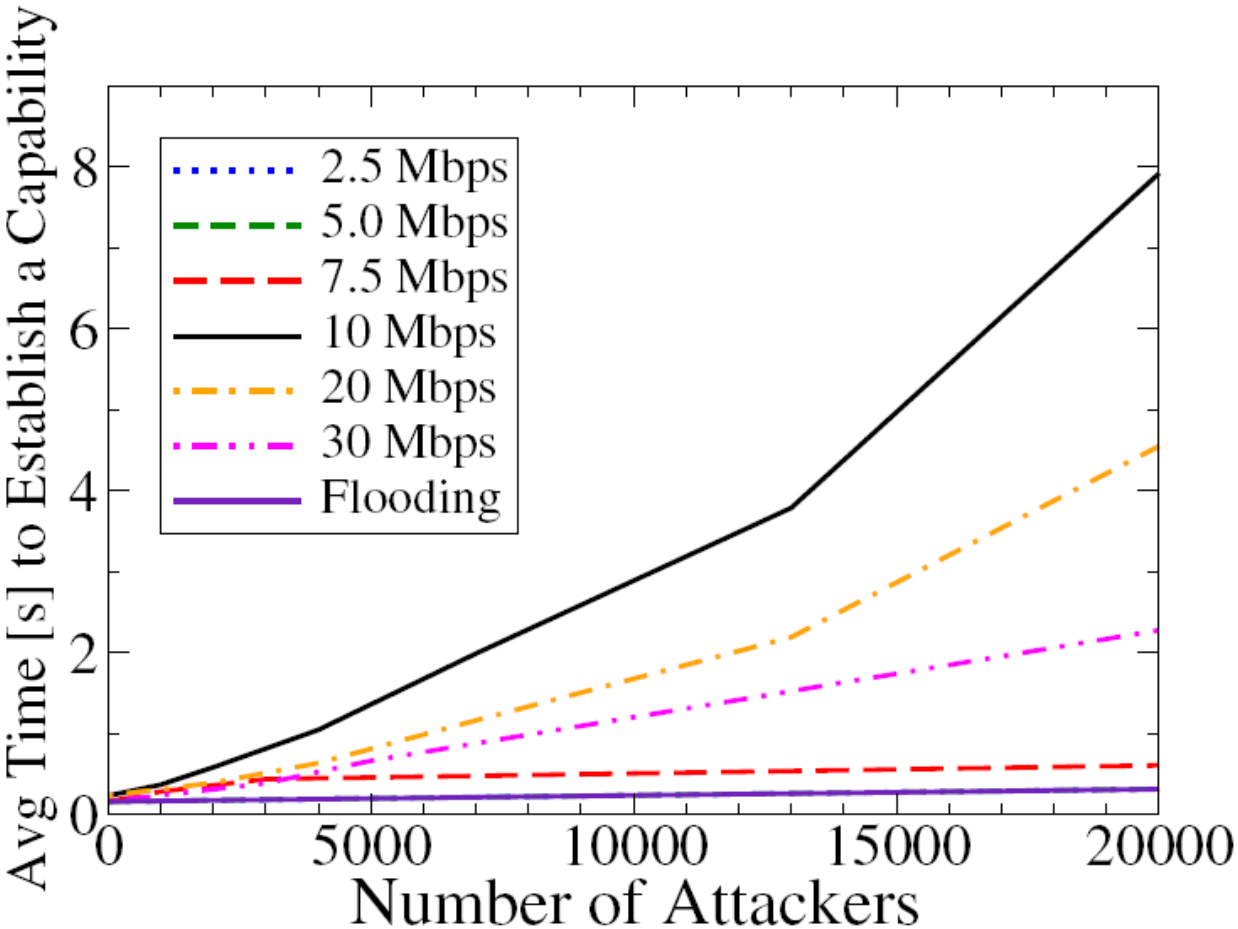
# Legitimate Sender Strategy



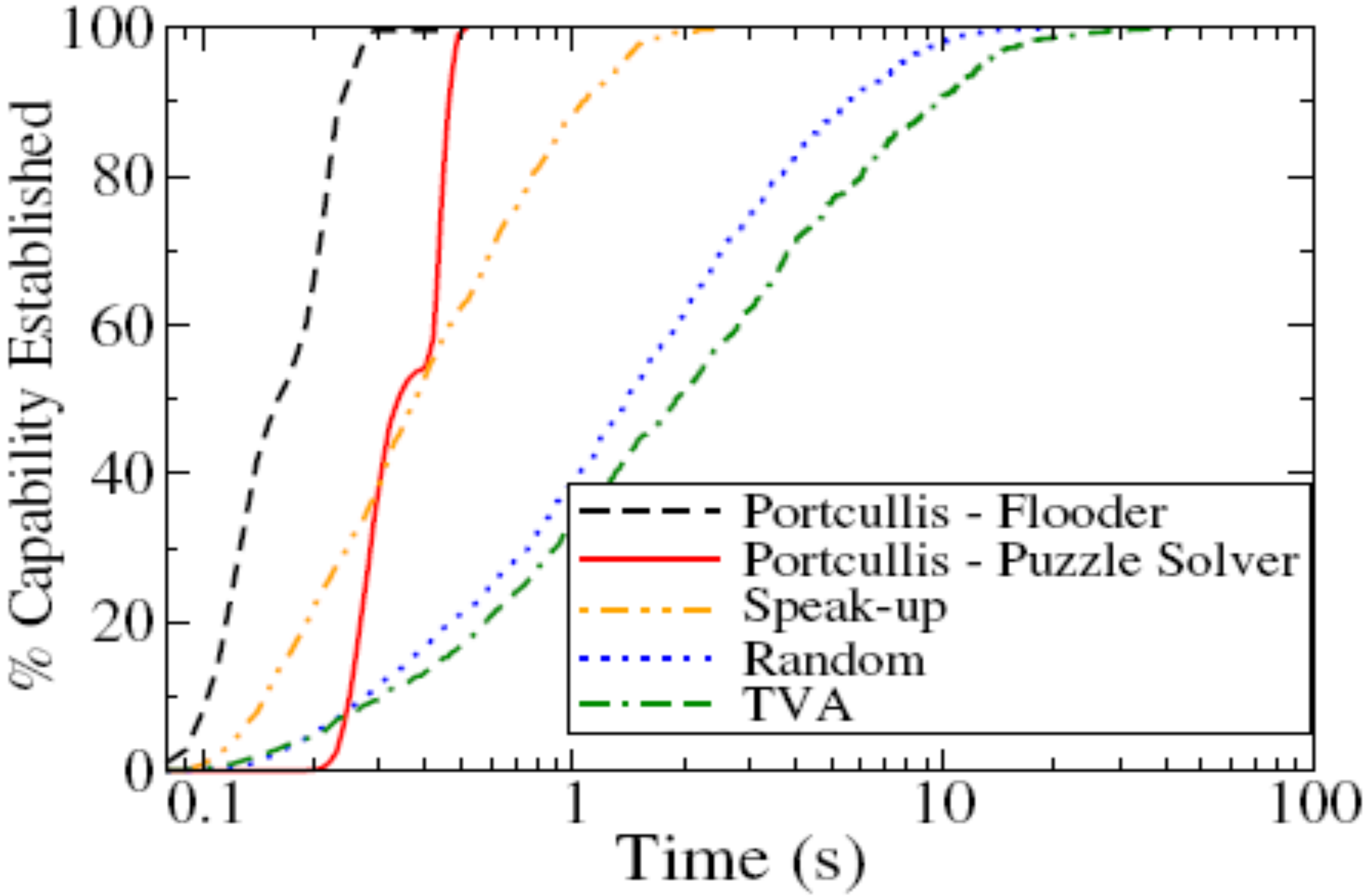
# Legitimate Sender Strategy



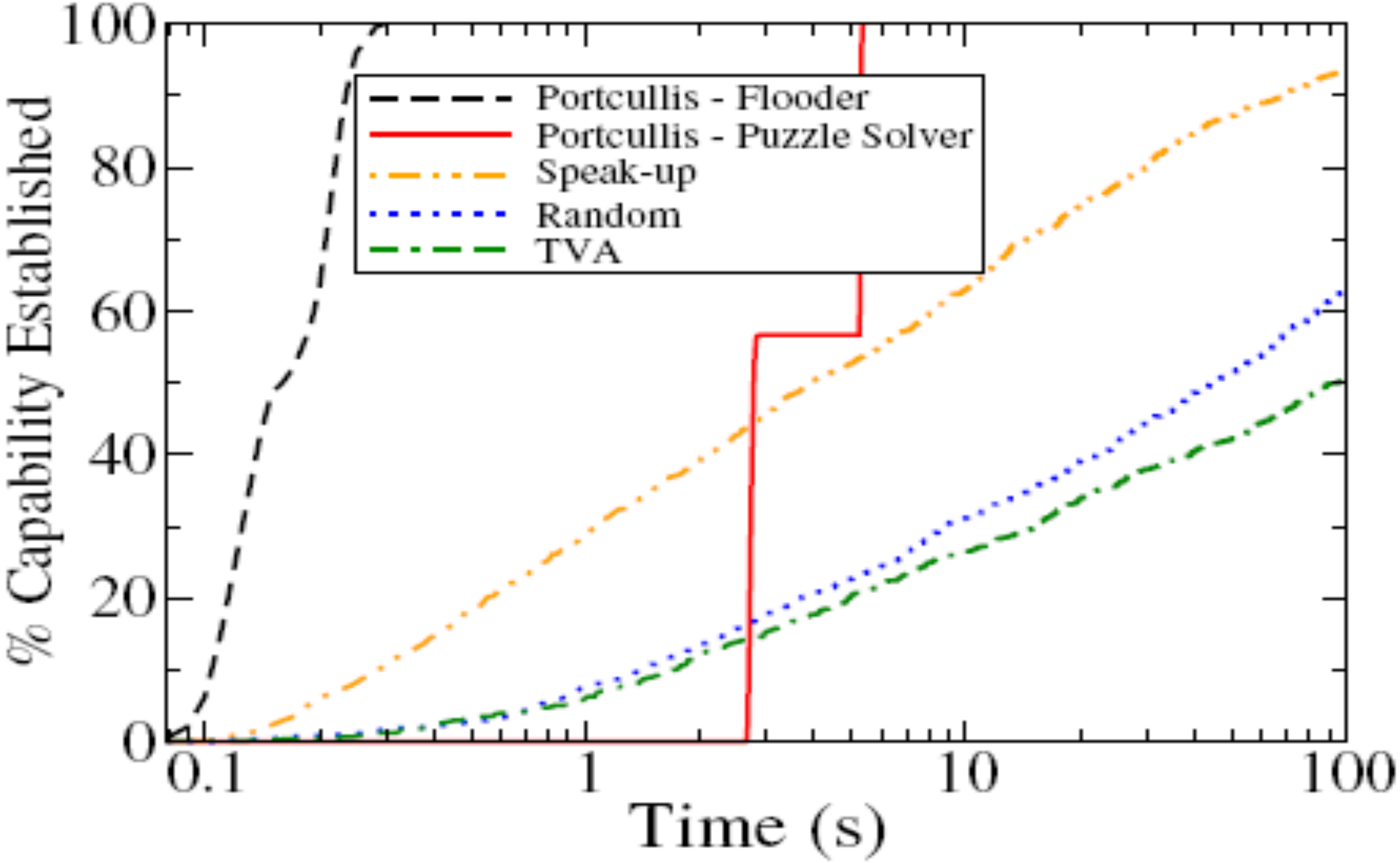
# Evaluation - Portcullis Attacker Strategies



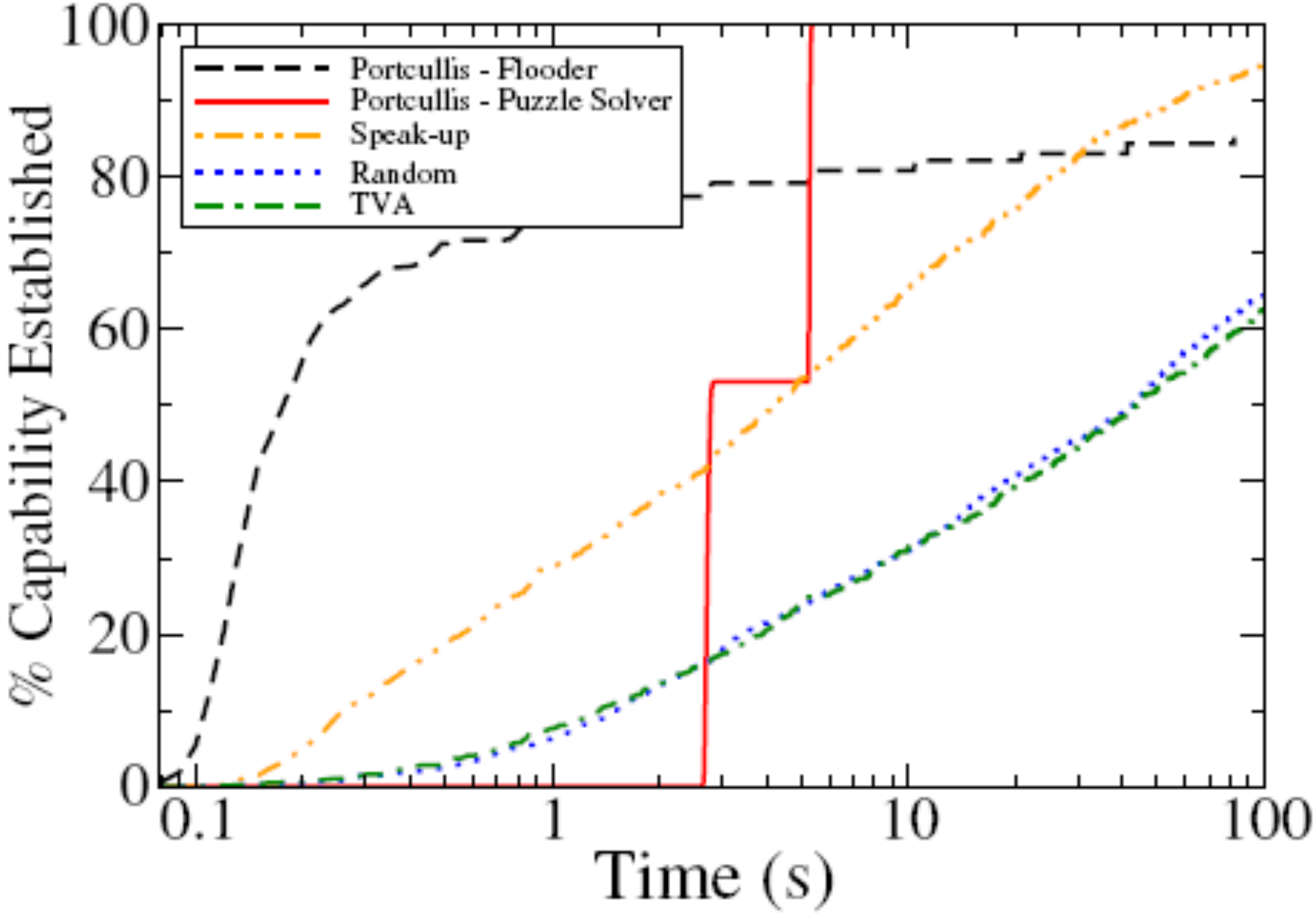
# Evaluation – Comparative Results



# Evaluation – Comparative Results (Cont’d)



# Evaluation – Comparative Results (Cont'd)



# Conclusion

- Portcullis mitigates DoC attacks by allocating bandwidth based on per-computation fairness.
- Novel puzzle mechanism strictly bounds the setup delay imposed by a given number of attackers.
- Makes capability systems a robust defense against DDoS attacks.