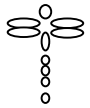


Prevent DoS using IP source address spoofing

MATSUZAKI ‘maz’ Yoshinobu
<maz@iij.ad.jp>

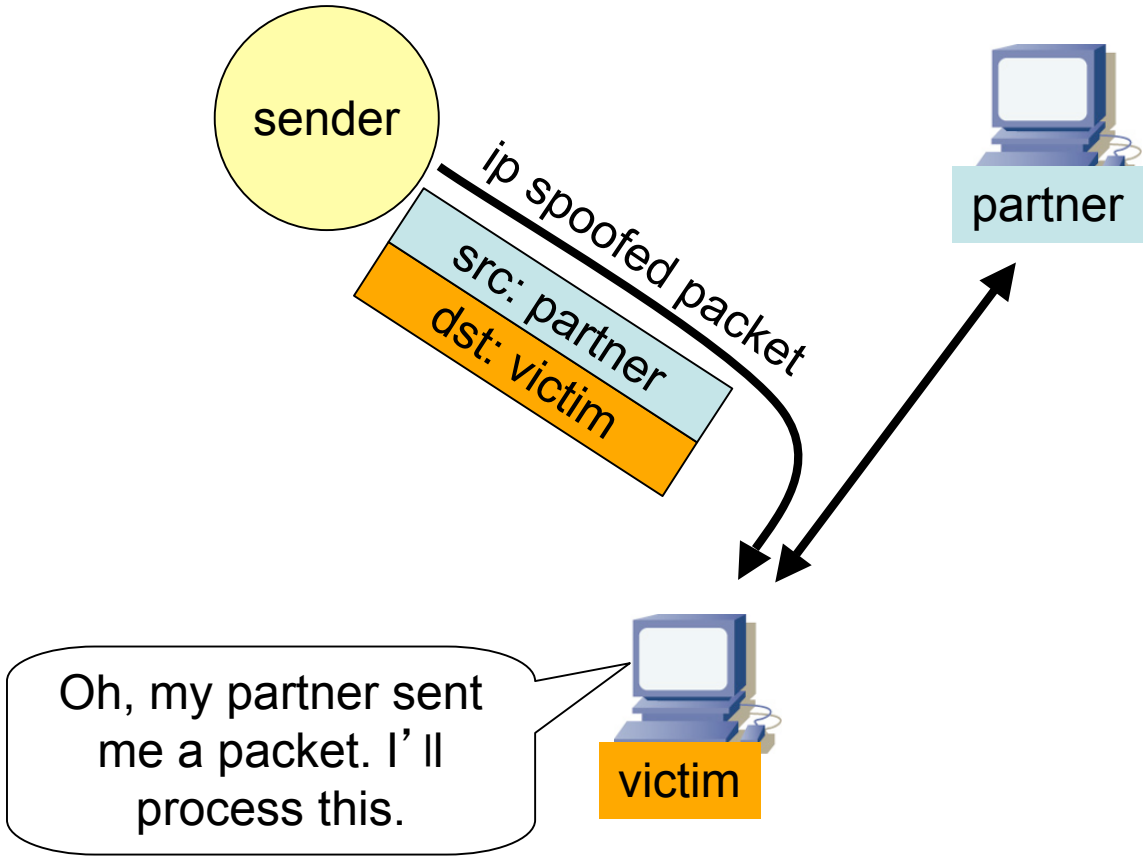
ip spoofing

creation of IP packets with source addresses other than those assigned to that host

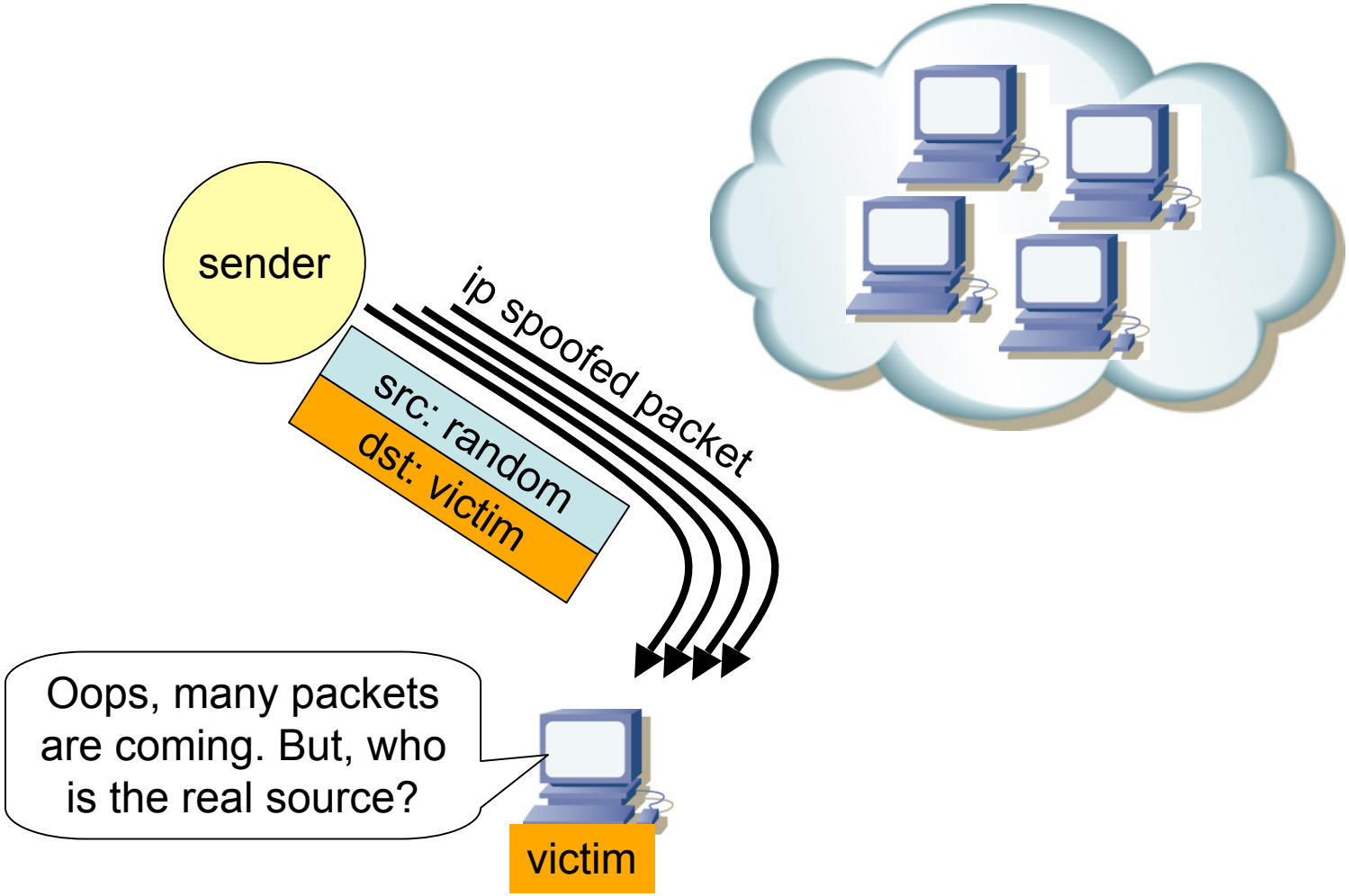
Malicious uses with IP spoofing

- impersonation
  - session hijack or reset
- hiding
  - flooding attack
- reflection
  - ip reflected attack

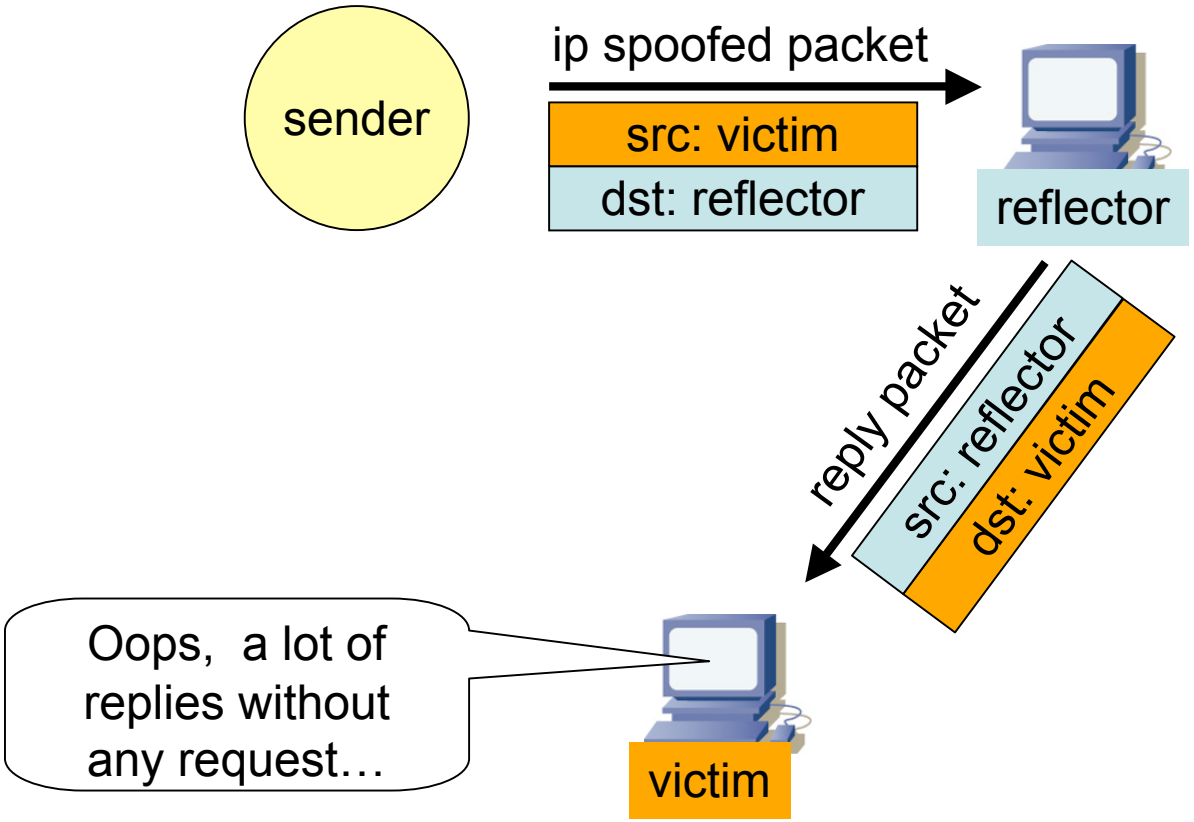
impersonation



hiding



reflection

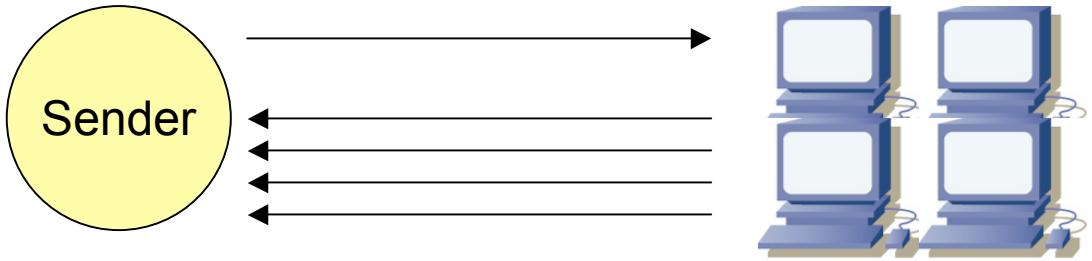


ip reflected attacks

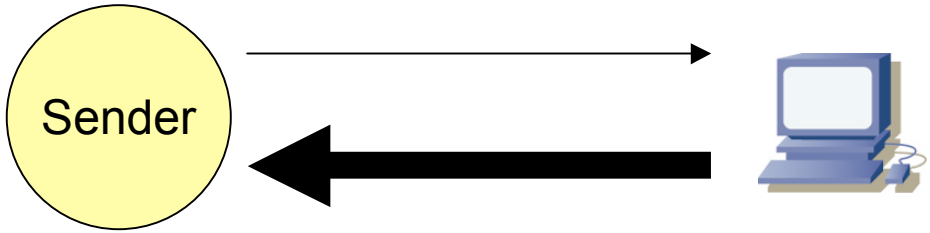
- smurf attacks
  - icmp echo (ping)
  - ip spoofing(reflection)
  - amplification(multiple replies)
- **dns amplification attacks**
  - **dns query**
  - **ip spoofing(reflection)**
  - **amplification(bigger reply/multiple replies)**

amplification

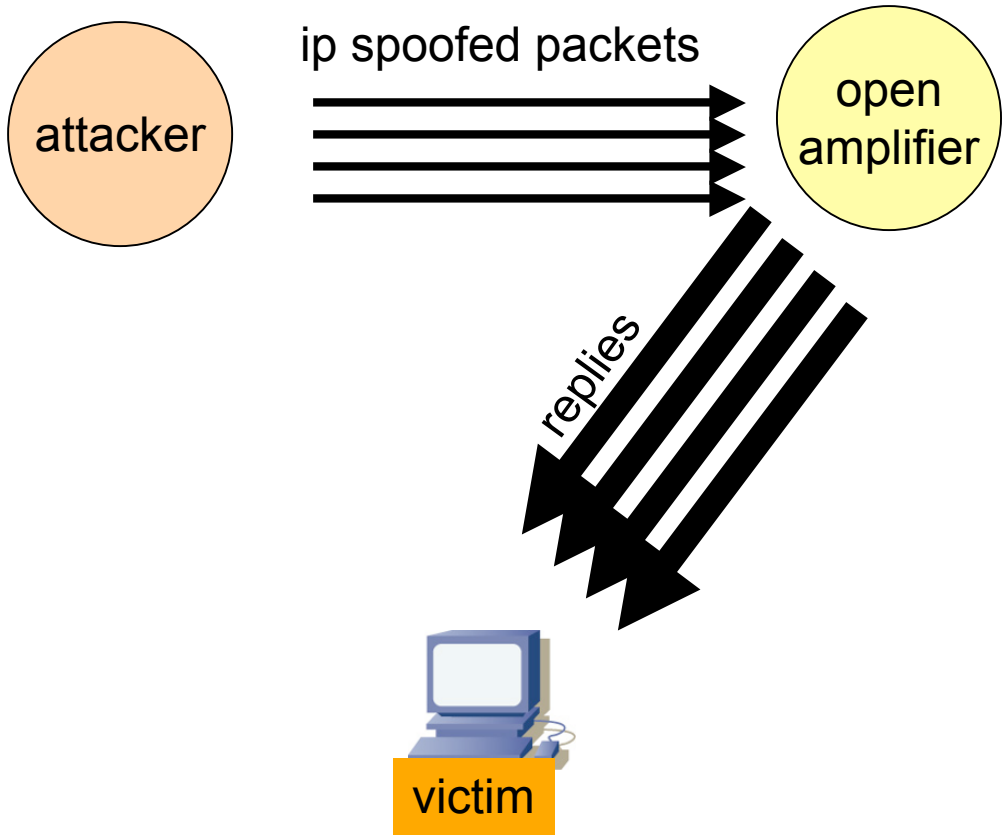
1. multiple replies



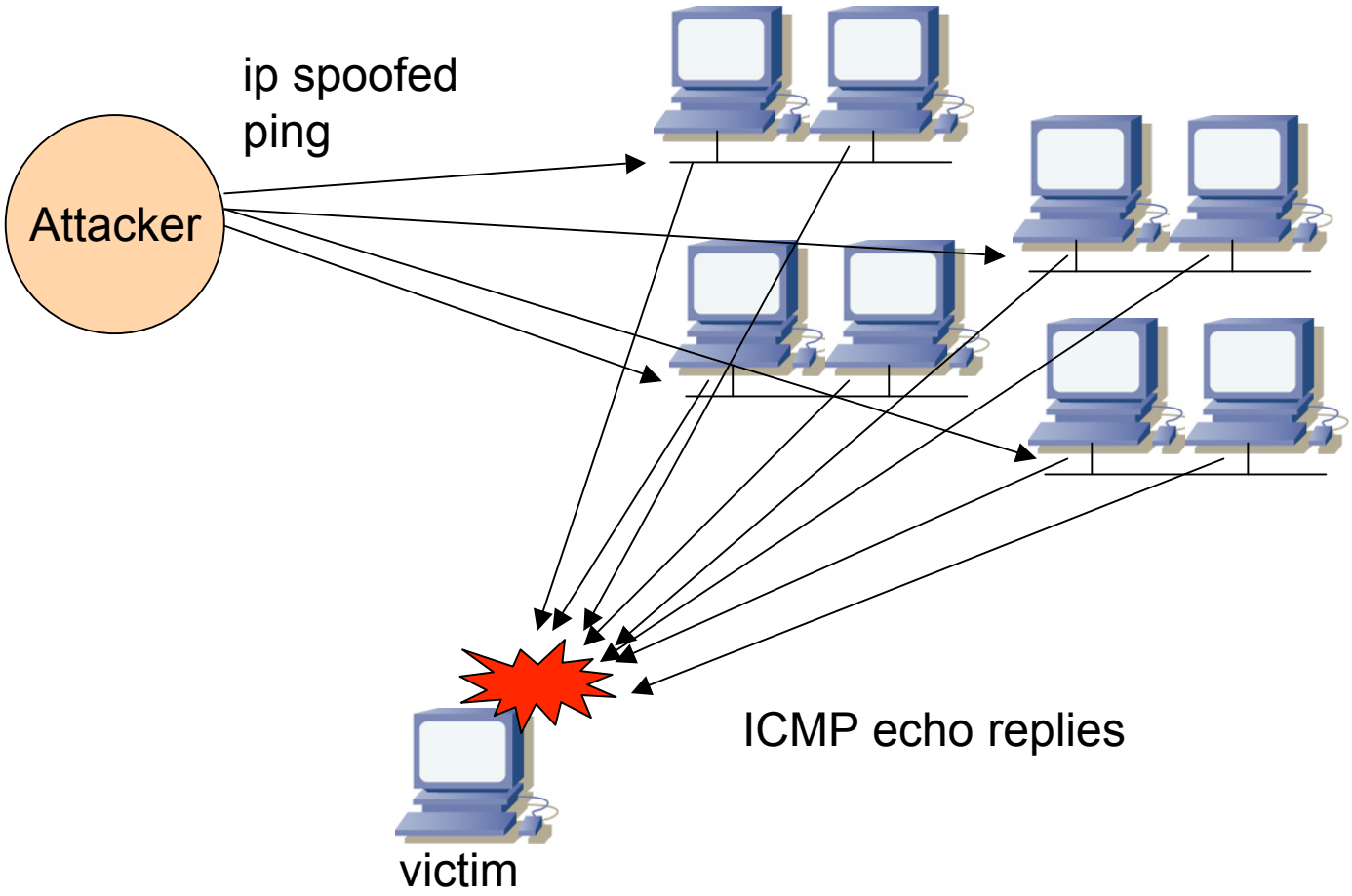
2. bigger reply



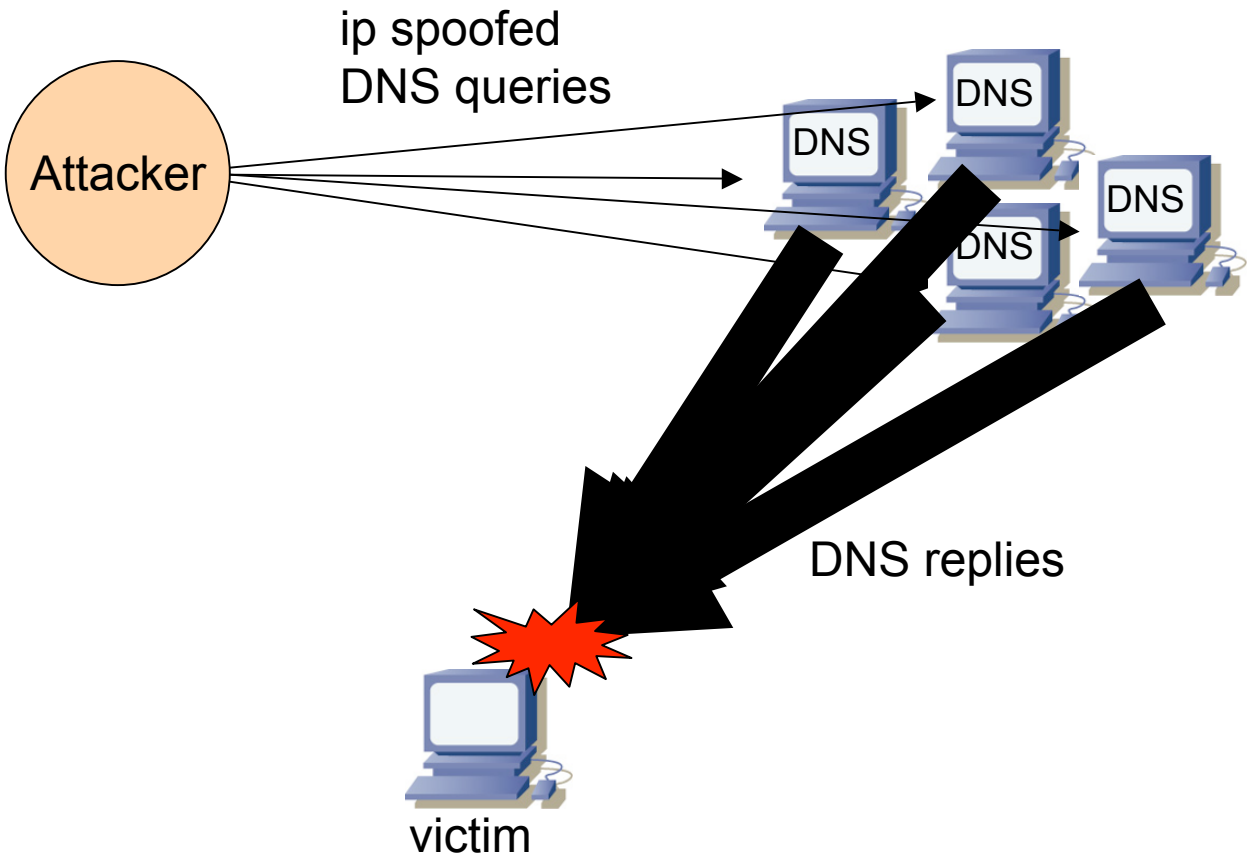
ip reflected attacks



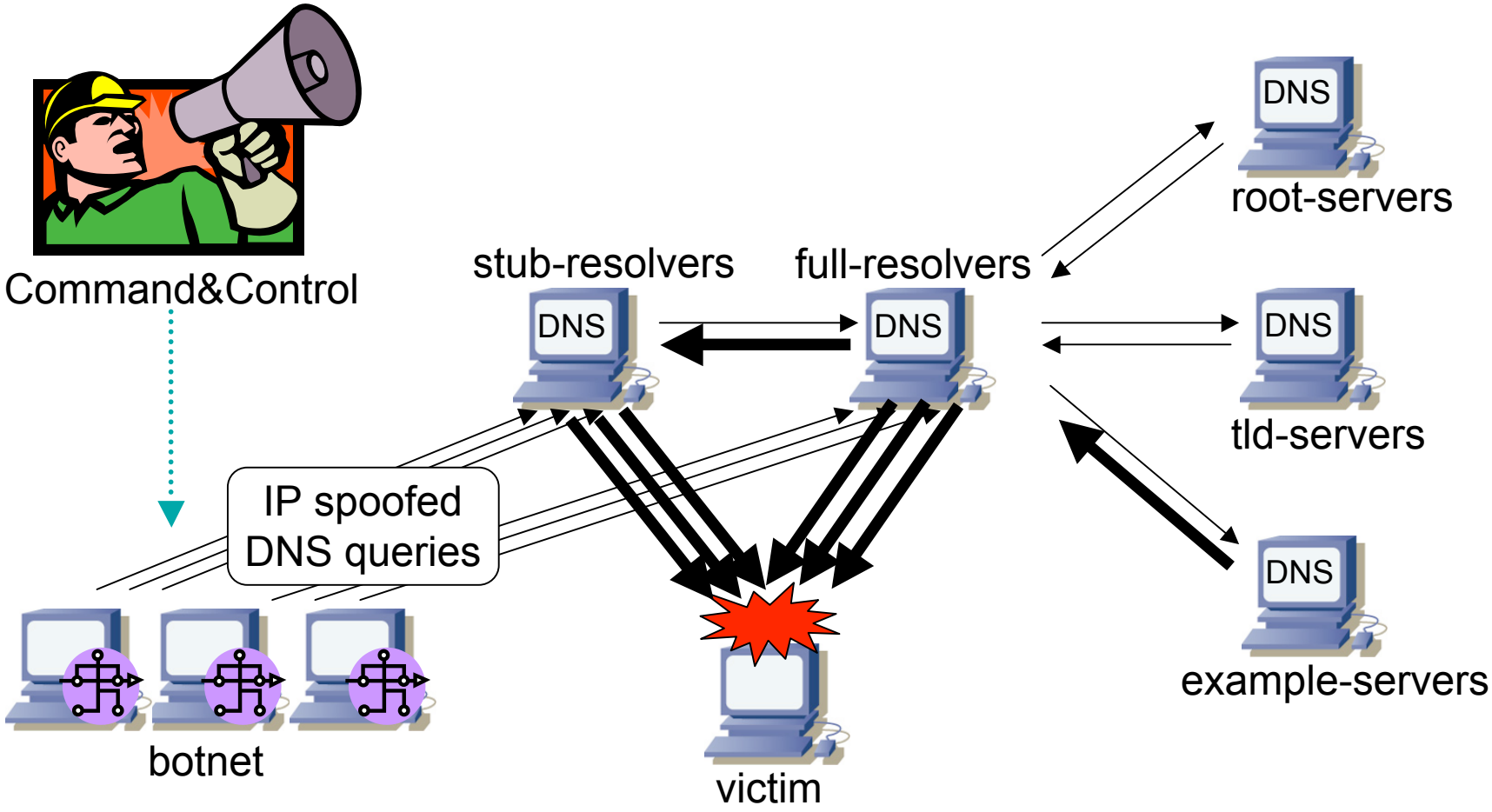
smurf attack



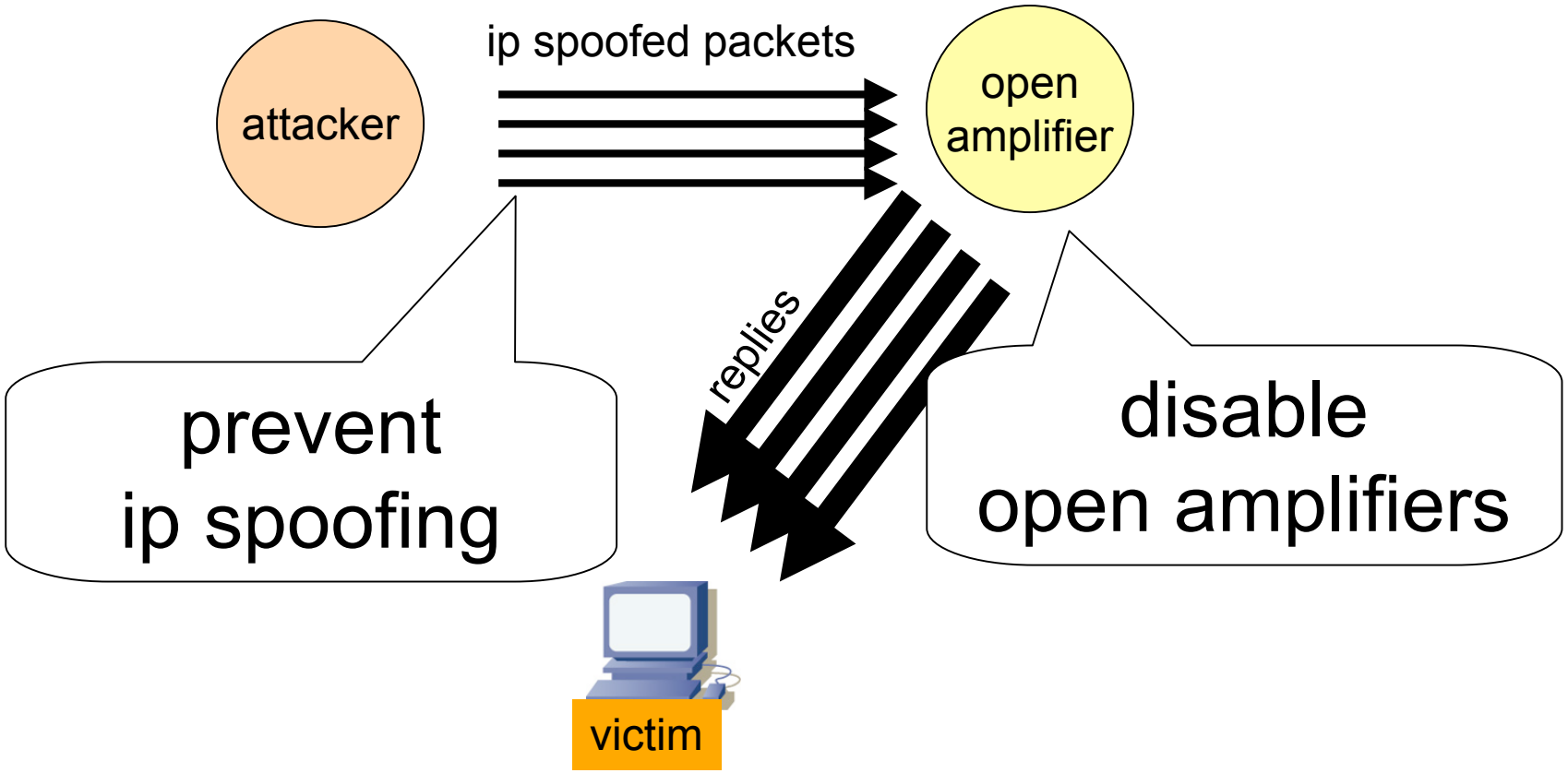
dns amplification attack



relations – dns amp attack



solutions for ip reflected attacks



two solutions

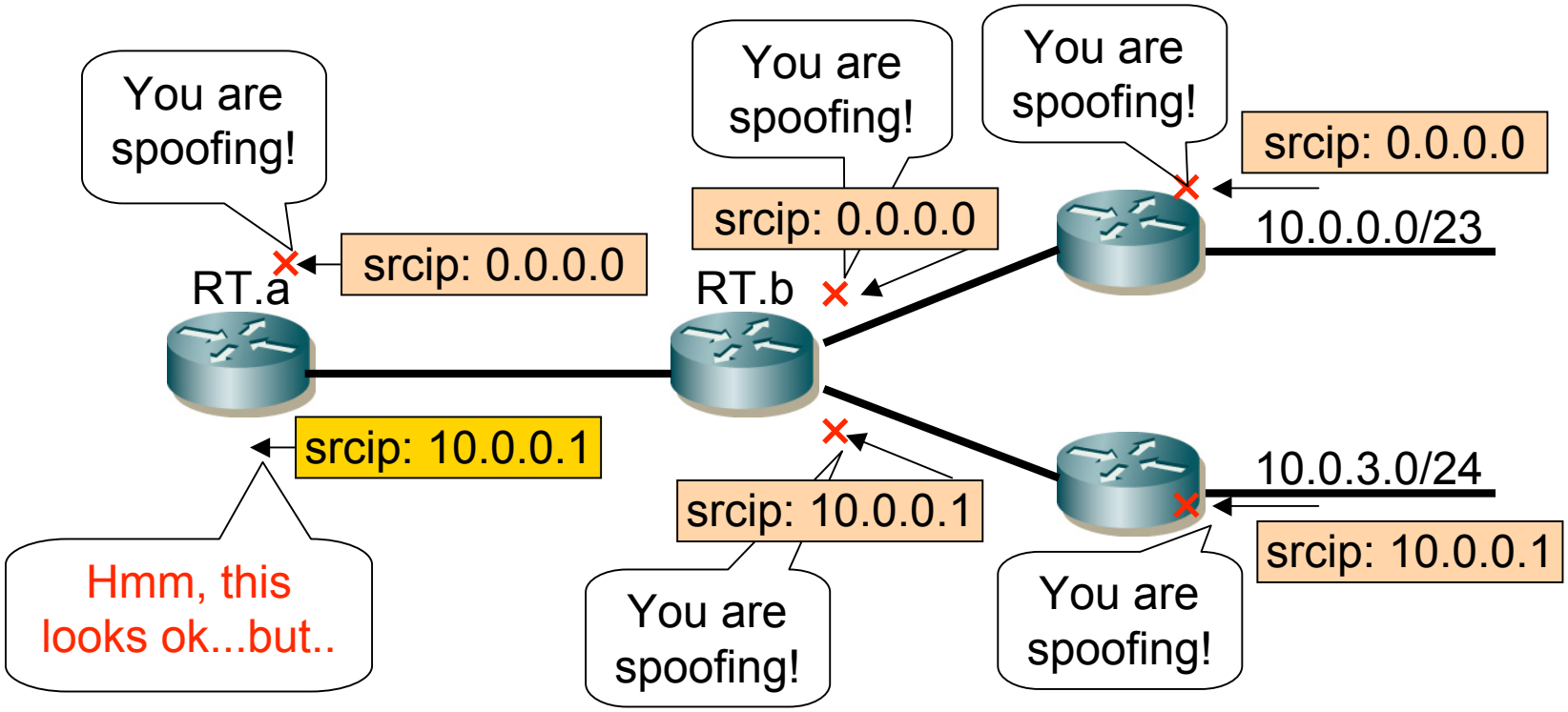
- disable ‘open amplifier’
  - disable ‘directed-broadcast’
  - disable ‘open recursive DNS server’
    - contents DNS server should accept queries from everyone, but service of resolver (cache) DNS server should be restricted to its customer.
- prevent ip spoofing!!
  - source address validation
  - BCP38 & BCP84

Source Address Validation

- Check the source ip address of ip packets
  - filter invalid source address
  - filter close to the packets orign as possible
  - filter precisely as possible

- If no networks allow ip spoofing, we can eliminate these kinds of attacks

close to the origin

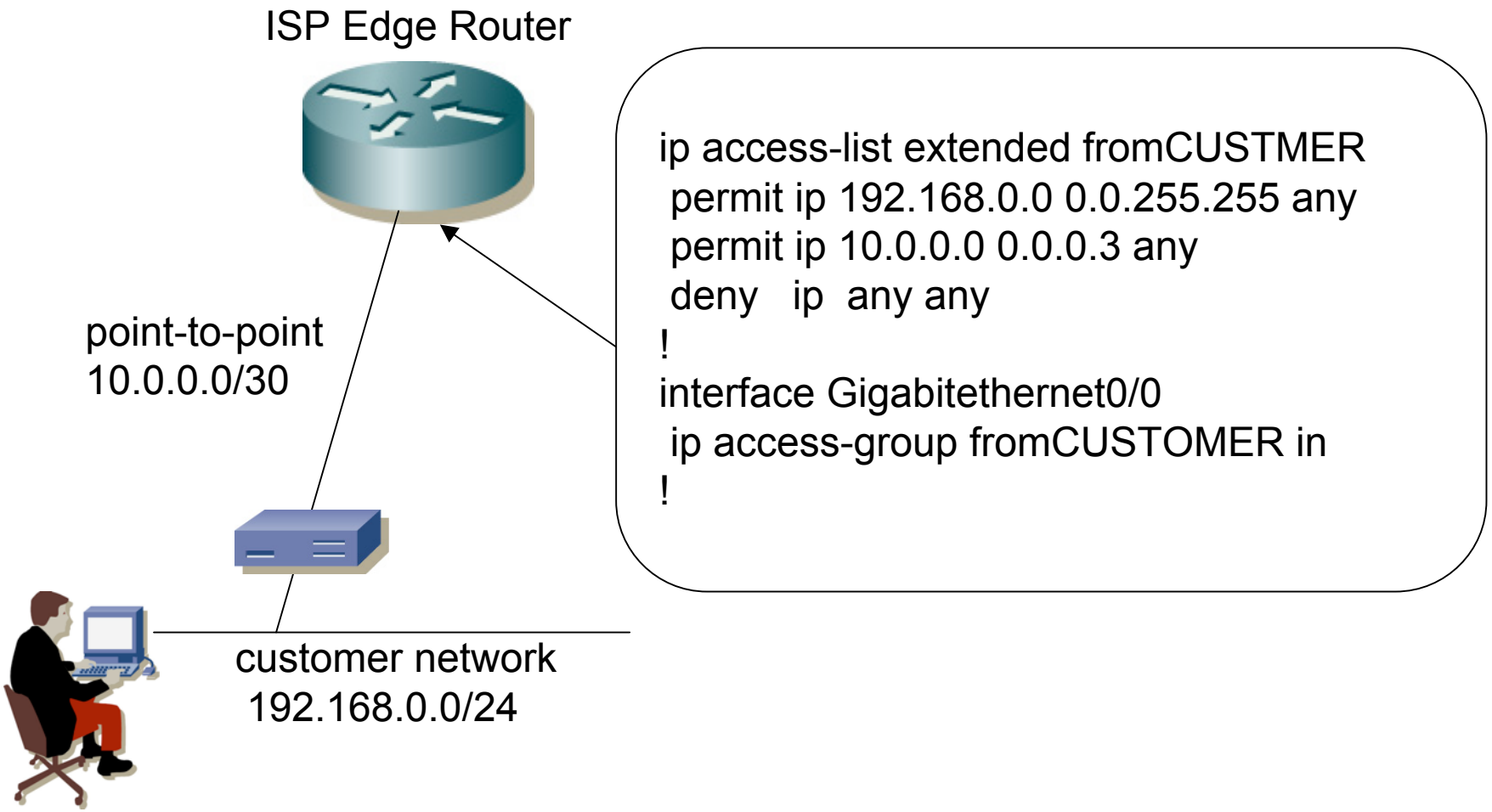


- we can check and drop the packets which have unused address everywhere, but used space can be checked before aggregation

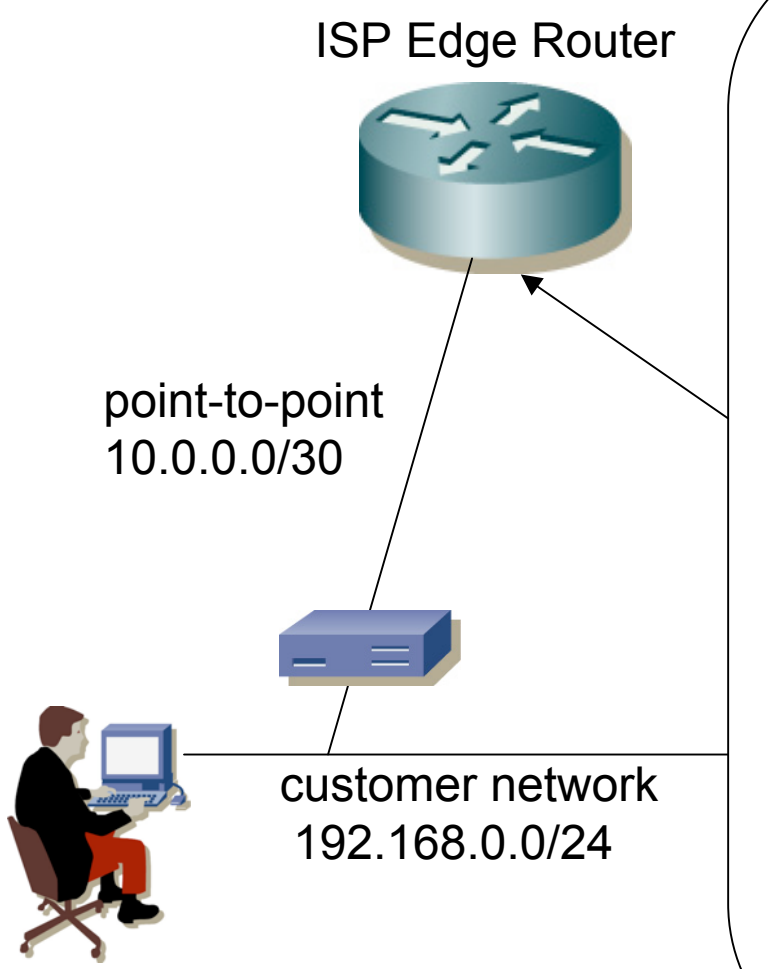
how to configure the checking

- ACL
  - packet filter
  - permit valid-source, then drop any
- uRPF check
  - check incoming packets using ‘routing table’
  - look-up the return path for the source ip address
  - loose mode can’t stop ip reflected attacks
    - use strict mode or feasible mode

cisco ACL example

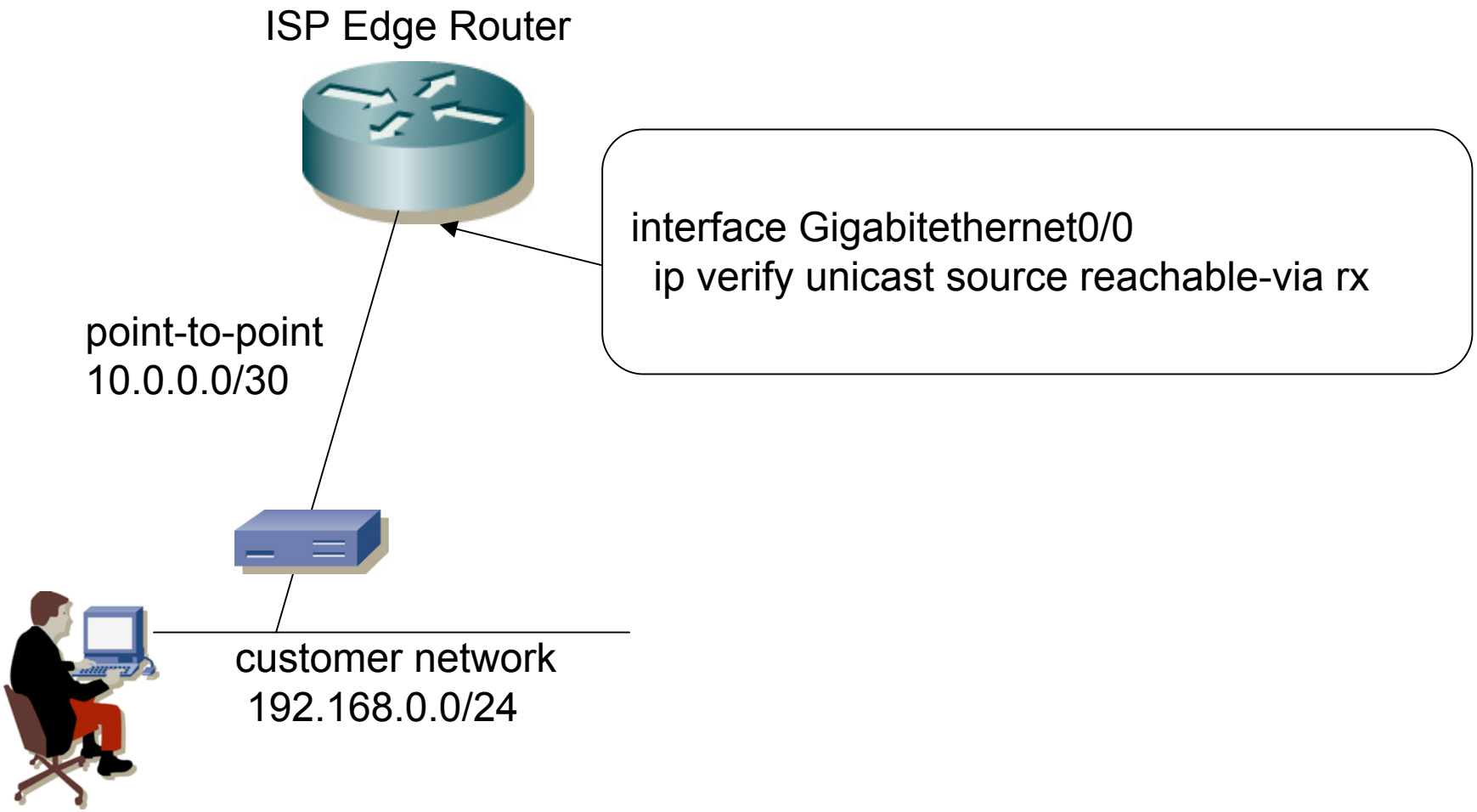


juniper ACL example

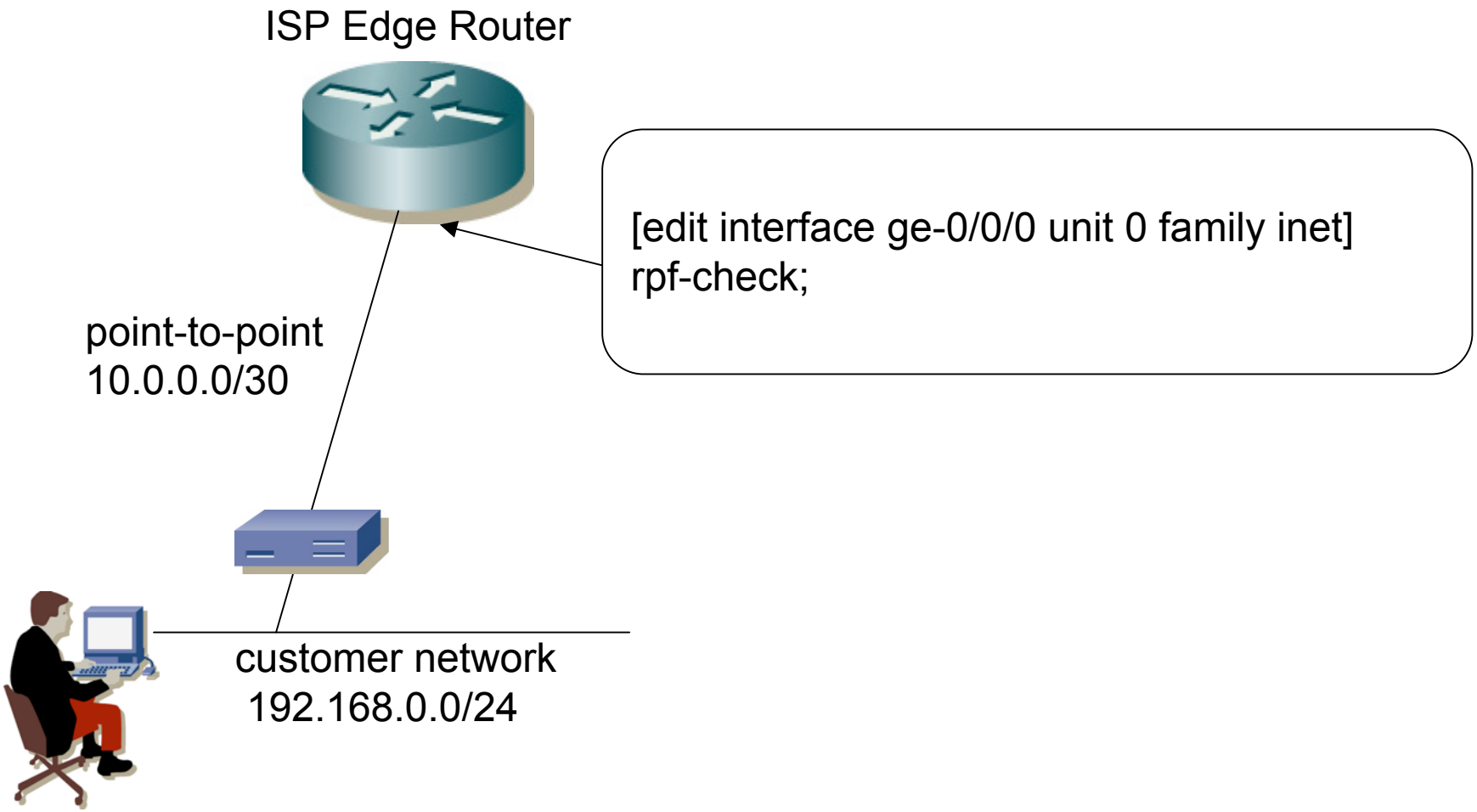



```
firewall family inet {
 filter fromCUSTOMER {
  term CUSTOMER {
   from source-address {
    192.168.0.0/16;
    10.0.0.0/30;
   }
   then accept;
  }
  term Default {
   then discard;
  }
 }
}
[edit interface ge-0/0/0 unit 0 family inet]
filter {
 input fromCUSTOMER;
}
```

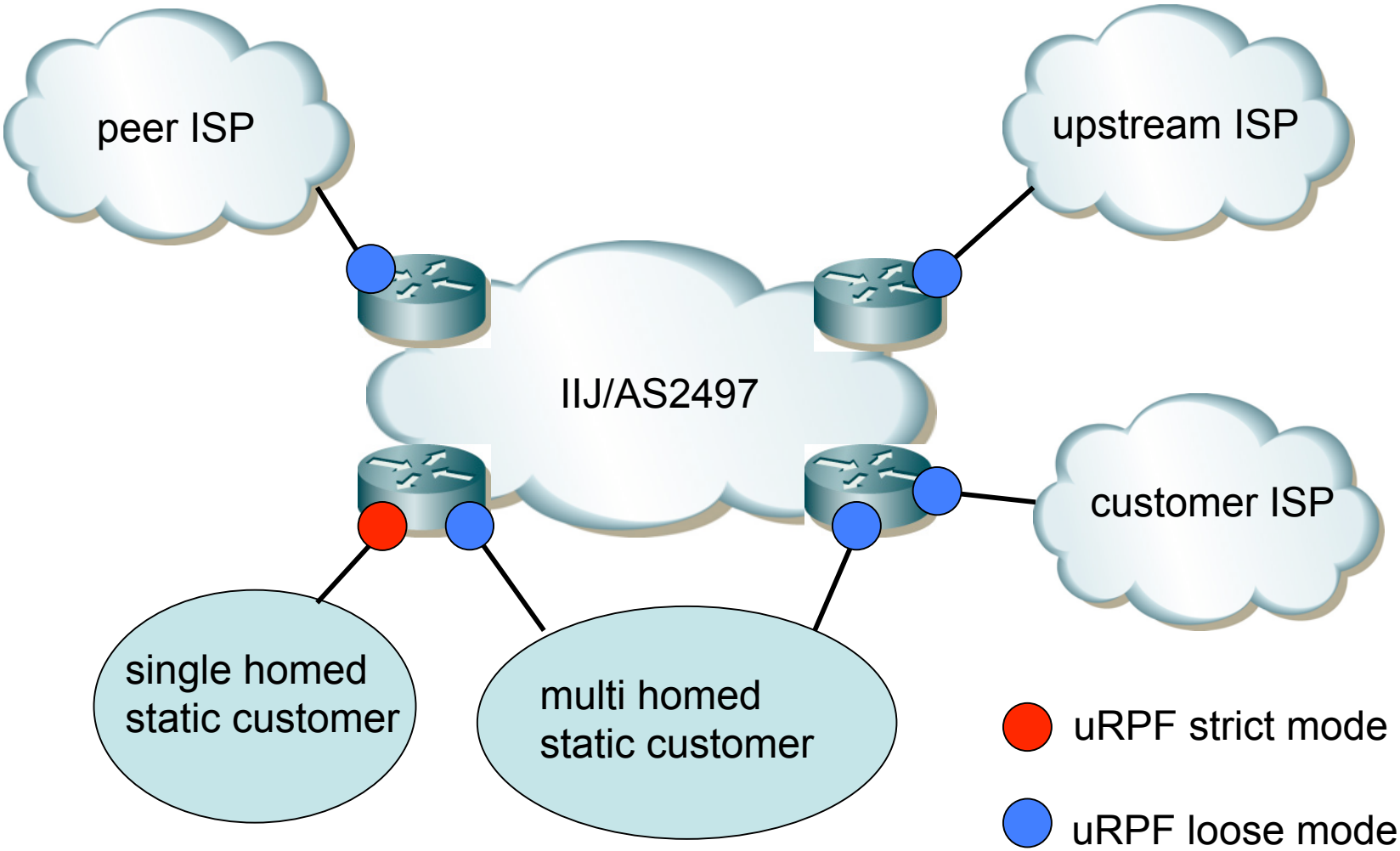
cisco uRPF example



juniper uRPF example



IIJ's policy



ACL and uRPF

- ACL
  - deterministic ☺
    - statically configured
  - maintenance of access-list ☹
- uRPF
  - easy to configure ☺
  - care about asymmetric routing ☹
    - strict mode is working well only for symmetric routing
    - loose mode can’t stop the ip reflected attack
    - there is no good implementation of feasible mode

END