



Eclipse Attacks - Hardening the P2P Layer

Hardening the P2P overlay network

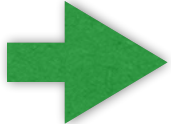
1. Why not smaller static timeouts?
2. Why not requesting from multiple peers?
3. What about alternative relay networks?

Security vs Scalability tradeoffs

Hardening the P2P overlay network

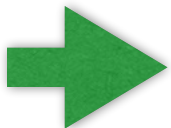
1. Dynamic timeouts

Hardening the P2P overlay network

1. Dynamic timeouts
 2. Handling Transaction Advertisements
 - **Filtering by IP address**
 - Randomly choosing sender
-  First request from one peer, then two, then three...

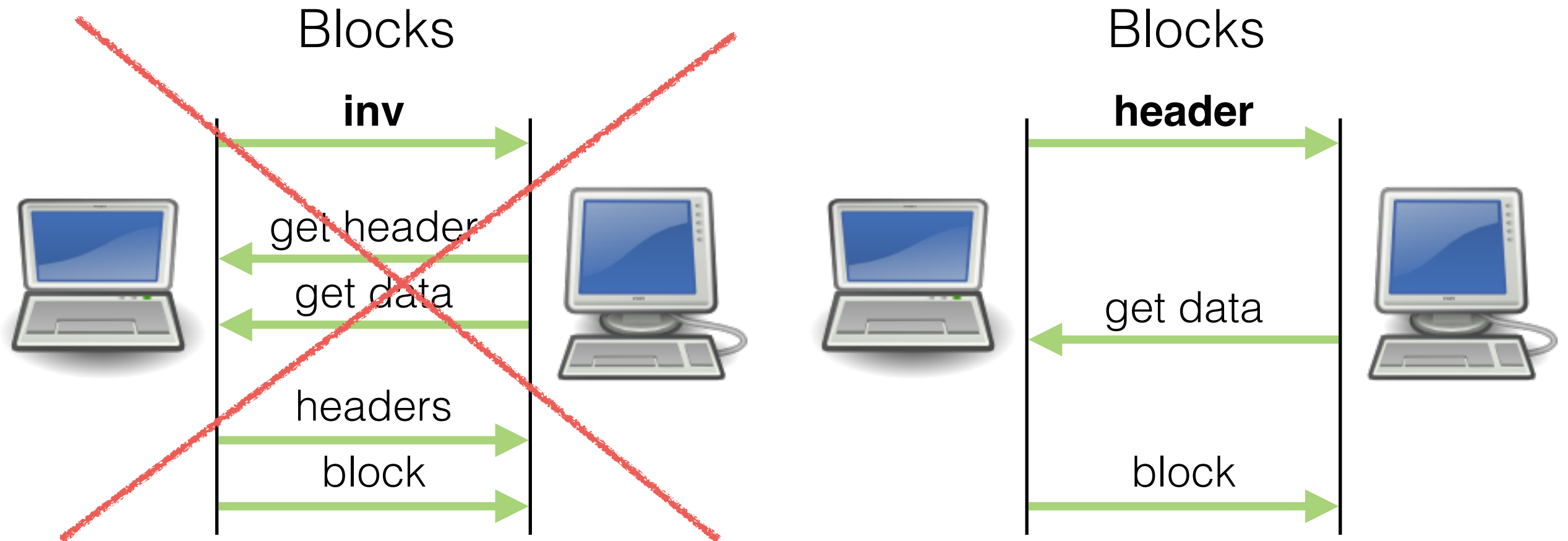
Hardening the P2P overlay network

1. Dynamic timeouts
2. Handling Transaction Advertisements
 - **Filtering by IP address**
 - Randomly choosing sender

 First request from one peer, then two, then three...
3. Updating Block Advertisements:
 - **Broadcast header instead of hash**
 - Keep track of block advertisers

Hardening the P2P overlay network

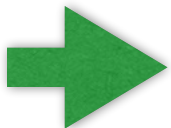
Better Block request management



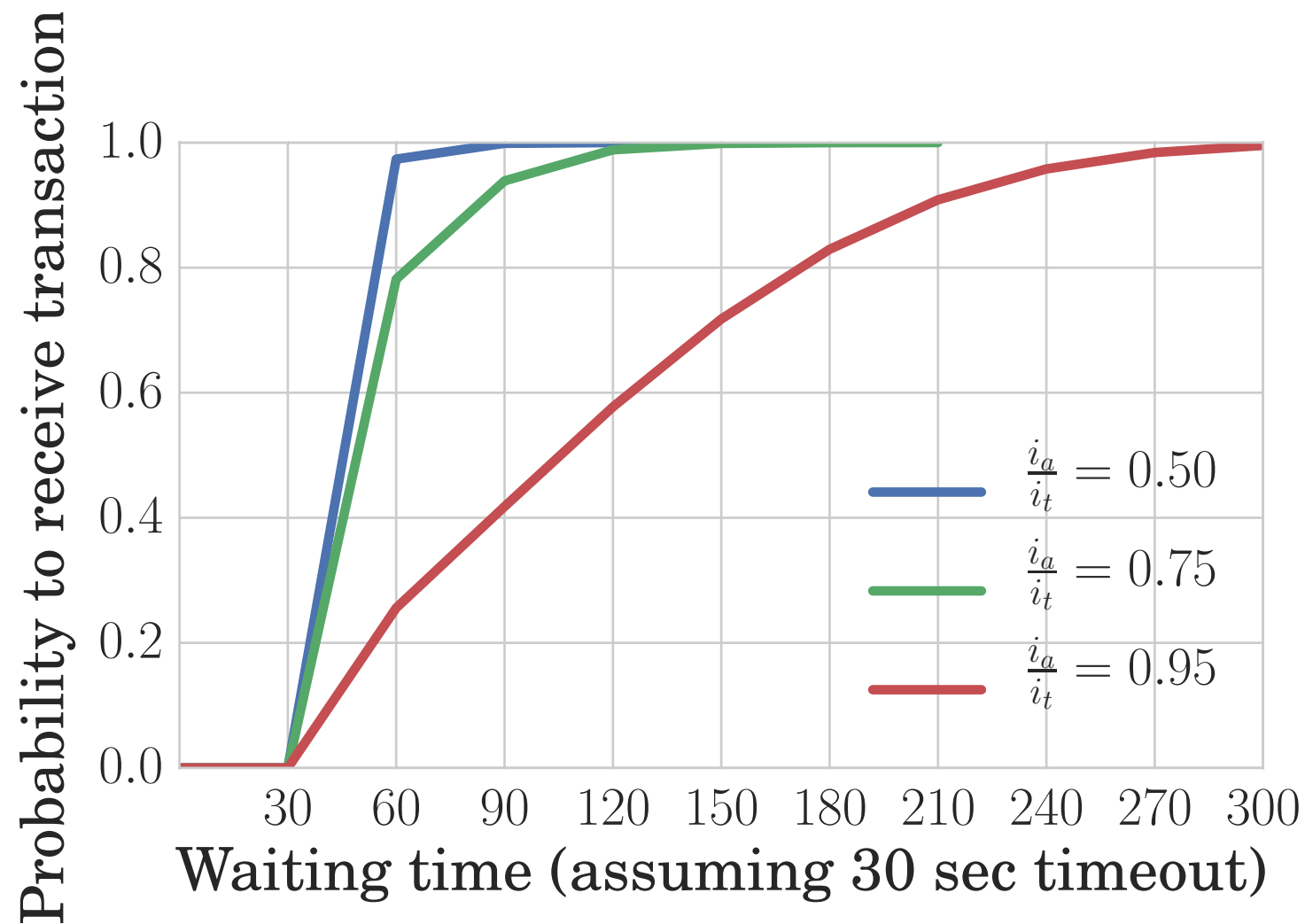
size(inv) = 36 bytes
size(header) = 80 bytes

Hardening the P2P overlay network

1. Dynamic timeouts
2. Handling Transaction Advertisements
 - **Filtering by IP address**
 - Randomly choosing sender

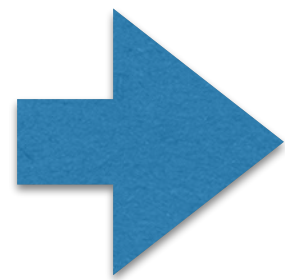
 First request from one peer, then two, then three...
3. Updating Block Advertisements:
 - **Broadcast header instead of hash**
 - Keep track of block advertisers

Hardening the P2P overlay network



i_a = **inv** messages sent by adversary

i_t = total **inv** messages



After 5 minutes, transaction is received, even if the adversary controls 95% of the inv