Peer finding protocol initial proposal

March 27, 2021

1 Vocabulary

1. Peers

   Entities using this protocol to discover and connect with one another

2. Discovery servers

   Entities facilitating the connection of peers via this protocol

3. Peer request

   A UDP datagram that is structured according to this protocol. A peer request is sent from a peer to a discovery server with the intent of obtaining information about other peers from the discovery server.

4. Peer response

   A UDP datagram that is structured according to this protocol. A peer response is sent by the discovery server to a peer in response to a peer request.

5. NAT

   Network Address Translation

2 Purpose

To allow peers who may be separated by arbitrary NAT layers to discover and form connections with each other.
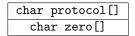
3 Structure

1. Network

   Any number of peers and any number of discovery servers may operate on the same network.

2. Peer request

Note: The following structure has no padding. Every string contained within is ASCII encoded and null terminated.

char protocol[]
char zero[]

A peer request begins with a string that names the protocol on which the sending peer is seeking peers. The protocol name is followed by an arbitrary number of null bytes. The length of the datagram will determine the number of addresses that will be given by the response, as the discovery server will respond with a datagram that is at most as large as the request datagram that prompted it. This is to prevent discovery servers from being exploited as a DDOS network using IP address spoofing, as has happened in old versions of the NTP protocol.

3. Peer response

Note: The following structure has no padding. Every string contained within is ASCII encoded and null terminated.

char protocol_name[]
struct { char address[], port[] } peers[]

A peer response begins with a string that names the protocol for which the datagram is a response. The protocol name is followed by pairs of strings which indicate peers. The first string in a pair gives the peer's network address, and the second gives its port.

4   Function

A discovery server will listen for peer requests and maintain a log of recent peers on a given named protocol. When a discovery server receives a request, it will fill a response packet with the name of the protocol specified by the request, along with some number of peer addresses/port pairs, and send the response back to the address and port it received the request from. A request packet that is only as large as the name of the protocol will prompt a response that only contains the name of the protocol. This case can be used by a peer to test mutual visibility with the server, and as a keepalive for UDP holepunching.

A peer should begin with a list of one or more discovery servers. It will use these servers to request address/port information of additional peers known by the servers.

5 Advice

1. A discovery server should ideally respond with peers that are known to use the same protocol as the request. Peers should be able to determine if they can communicate on their own, so if protocol matching is not done, it's not the end of the world, but it will probably be less efficient.

2. A peer should send out request packets reasonably often in order to keep a UDP hole through any NAT layers open and to appear active to discovery servers.