ZeroTier Protocol

Monica Moniot

January 19, 2023

1 Definitions

For clarity's sake, when I use the name ZeroTier, unless otherwise specified, I am referring to the V1 ZeroTier protocol.

Definition 1 (Node)**.** A *node* in ZeroTier is a client machine running the ZeroTier protocol. A node is capable of making peer-to-peer, end-to-end encrypted ethernet connections with any other node, so long as it knows the address of the other node, and the other node chooses to respond to its connection. Public keys are used to identify and authenticate nodes.

Since all nodes are capable of communicating with each other, they form a global virtual ethernet network.

Definition 2 (Address)**.** The *address* of a node in ZeroTier is a uniquely identifying 40-bit number derived from the hash of the node's public key.

Definition 3 (Identity)**.** The *identity* of a node in ZeroTier consists of that node's public key and the matching address.

Definition 4 (Path)**.** A *path* is the physical link that is used for one node to send network packets to another. Generally this is a direct link over UDP, in which case the path is fully specified by the IP address and open port of the destination node. Paths are automatically found and established using DNS-like name resolution and NAT hole punching.

Definition 5 (Network Identifier)**.** A *network identifier* is a 64-bit number consisting of an address followed by a 16-bit "network number".

Definition 6 (Network Controller)**.** A *network controller* is simply a node that has had it's address distributed as part of a network identifier. Network controllers are able to define a sub-network of the ZeroTier global virtual network. Nodes are members of this network if and only if they have been given permission to join it by the network controller. The network controller is also able to distribute certificates, credentials and network rules to members of its network, giving it fine control over all interactions between nodes. It is not however able to decrypt or modify traffic over its network that was not directly sent to it.

Such networks create their own virtual IP address space; When a node joins a network, it is assigned a virtual IP address on this network, and other nodes on the same network can use that IP address to contact it. The network controller decides how IP addresses are assigned, and whether to use IPv4 or IPv6. To the host machine running ZeroTier, a ZeroTier network looks and behaves as if it is a LAN or WAN that the machine is physically connected to.

Definition 7 (MAC address)**.** The *MAC address* of a node in ZeroTier is a 48-bit number consisting of the address of the node xor'ed with a network identifier. As such a single node may have multiple MAC addresses but each of these MAC addresses is unique to each ZeroTier network it is a member of. These are the virtual MAC addresses assigned to machines for the sake of ethernet simulation.

Definition 8 (Root Servers)**.** *Root servers* are nodes operated by the ZeroTier organization that centrally handle the job of mapping all global addresses to their matching identities, and mapping addresses to possible paths to the owner of that address. Within the protocol, they are responsible for public key distribution and for coordinating initial contact between newly connecting machines.

Once keys are distributed and contact is established nodes will no longer contact root servers. The only exception being if the path connecting two nodes goes down, and a new path needs to be established, or if for whatever reason a direct path between two nodes cannot be established, so the nodes decide to route their traffic through their shared connection to the root servers.

Network traffic in ZeroTier is peer-to-peer and end-to-end-encrypted. Once a secure session is established between two nodes, it is cryptographically infeasible for the ZeroTier organization to read or modify any node's network traffic. However, it is possible for the ZeroTier organization to perform a man-in-the-middle attack on newly connecting nodes that have not yet established a session. If a root server is compromised, when it is asked to map an address to its node's identity, the root can instead brute-force the 40-bits of the address to generate an identity that matches it. It can then respond with this identity it controls instead of the correct one. The root server must then sabotage the ZeroTier rendezvous protocol so that network traffic between these two nodes is only routed through the root servers. Due to root choice randomization this attack is significantly less successful if just one uncompromised root is online. Once all this is done a classical man-in-the-middle attack can follow. Any out-of-band confirmation that the identity of the node being contacted is the same as the identity returned by the root will render this attack impossible.

Due to security and reliability concerns it is not yet possible for independent administrators to operate their own root servers. However it is possible to operate mirrors of the root servers. These mirrors, referred to as "moons" create a local copy of some subset of the root server database, and will defer to the root servers if a request is made of them for an address not present in their database.

2 ZeroTier Rendezvous Protocol

All ZeroTier nodes start only knowing a few paths to the root servers. ZeroTier is able to bootstrap this into a peer-to-peer connection with any other node. Given the address of a node, a source node will contact a random root server to request the identity of the node with that address, the destination node. The root server will look up the destination node's identity and return it. Once the source node has its identity, it will then choose another random root server to request a rendezvous with the destination node. The root server will then search through all known paths to both the destination and the source nodes, and choose a pair most likely to succeed in back and forth communication. To the source node it will send the destination's path, and to the destination node it will send the source's path. Upon receipt both nodes will attempt to contact each other through the given paths. If either contact is successful the recipient will cache the successful path. From then on both node will choose the highest quality cached path to communicate with each other. Upon receipt of any packet from a node, the path it was received over is either added to the path cache, or updated in the cache to help with quality determination.

3 ZeroTier Client Protocol

When ZeroTier is installed into a host machine, it will generate a random public and private key. The private key is saved to the host machine. The public key is used to generate an address and identity. This identity is submitted to a random root server, and if the root server confirms that the node's address is not in use, it is saved to both the host machine and the root server. If it is in use the host machine will generate a new keypair and try again.

Next the host machine will ask for a network identifier from the user. When this is given the host machine will rendezvous with the network controller for that network. Upon a successful rendezvous the node will request to join the network controller's network. If the request is approved, the network controller assigns it a virtual IP address, and sends it any network-specific credentials or rules.

ZeroTier offers ARP and NDP emulation to help client machines map a virtual IP address on a given ZeroTier network to the ZeroTier address of its node. However these protocols are not emulated 1 to 1, in particular because ARP scales poorly on large networks. Instead, when a host machine wishes to connect to a new virtual IP address, it asks the network controller to unicast its request to connect directly to the owner of that virtual IP address, the destination node. If the destination node decides to connect, it will rendezvous with the host machine to established a peer-to-peer connection. Identities and paths are always

cached so in the future the host machine is able to resolve the IP address to a path to the destination node without the help of the network controller or a root server.