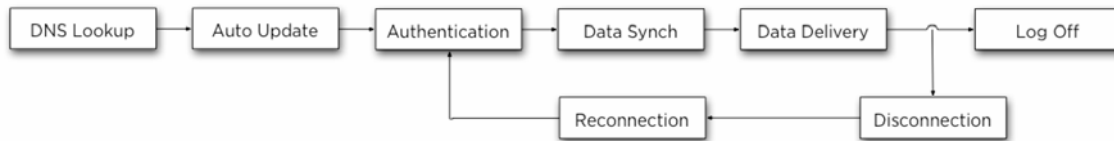


## Thinkpipes Network Configuration and Documentation

### Overview



**Figure 1: Typical Client Application Lifecycle**

The Thinkpipes Client application connects to a set of servers over the Internet to provide data and services to you, the end user. The client application goes through the following phases of activity during its lifecycle:

1. Server Lookup
2. Auto update
3. Login and authentication
4. Synchronization of Position and User data
5. Delivery of Data Updates
  - a. Delivery of Quote data
  - b. Delivery of Position data
  - c. Exchange of Order Status data
  - d. Delivery of Time and Sales data
  - e. Delivery of News data
  - f. Delivery of Charting data
6. (Possible loss of connectivity and reconnection)
7. Logoff and saving of data

Each of these phases may involve establishing various TCP connections to different destination ports. Depending on your corporation's ingress and egress policies for network traffic you may experience trouble with some or all of these phases.

This document attempts to explain the behavior of the Thinkpipes client application in terms of its network conversations. Later, we will discuss common network configuration scenarios that may apply to you and your company's network specifically so that you can configure our client application to work.

## Network Connection Information Summary

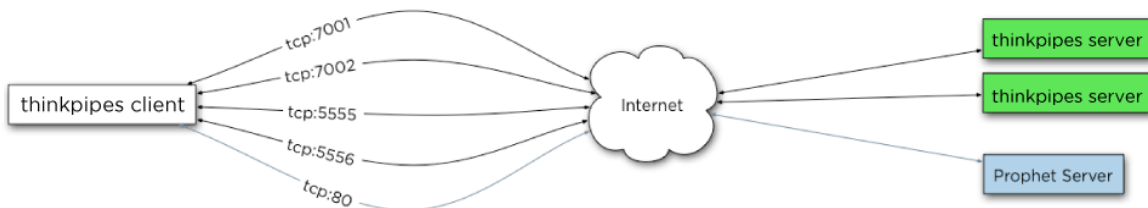
For those of you experienced in configuring your network the following information simply lists which ports are necessary for our application to work. All communication is initiated from the client side. The Thinkpipes client supports HTTP, HTTPS, SOCKs v4 and SOCKs v5 support and is configured from the "Options..." button in the login window. (SOCKs and HTTP/HTTPs tunneling support is currently undergoing testing)

Phase	Destination	Port
1. Server Lookup (DNS request)	your DNS resolver	UDP port 53
2. Auto Update	rainbow.thinkorswim.com	TCP port 7001
3. Login & Authentication	rainbow.thinkorswim.com	TCP port 7002
4. Synchronization of Data	rainbow.thinkorswim.com	TCP port 7002
5. Delivery of Data (except charts)	rainbow.thinkorswim.com	TCP port 5555, 5556
Delivery of Chart Data	207.33.55.212	TCP port 80
7. Reconnections	rainbow.thinkorswim.com	TCP port 7002
		TCP port 5555
		TCP port 5556

## Network Connection Details

Let's start by discussing the basic connection details regarding the Thinkpipes application.

In an ideal world, the following illustration identifies the primary components of the connectivity.



**Figure 2: Simplified view of Network connection**

### v v v

The simplified view of the network connectivity implies no inherent barriers to connecting from the client application to the servers. Each phase of the network lifecycle is detailed below.

#### 1. Server Lookup (client DNS lookup)

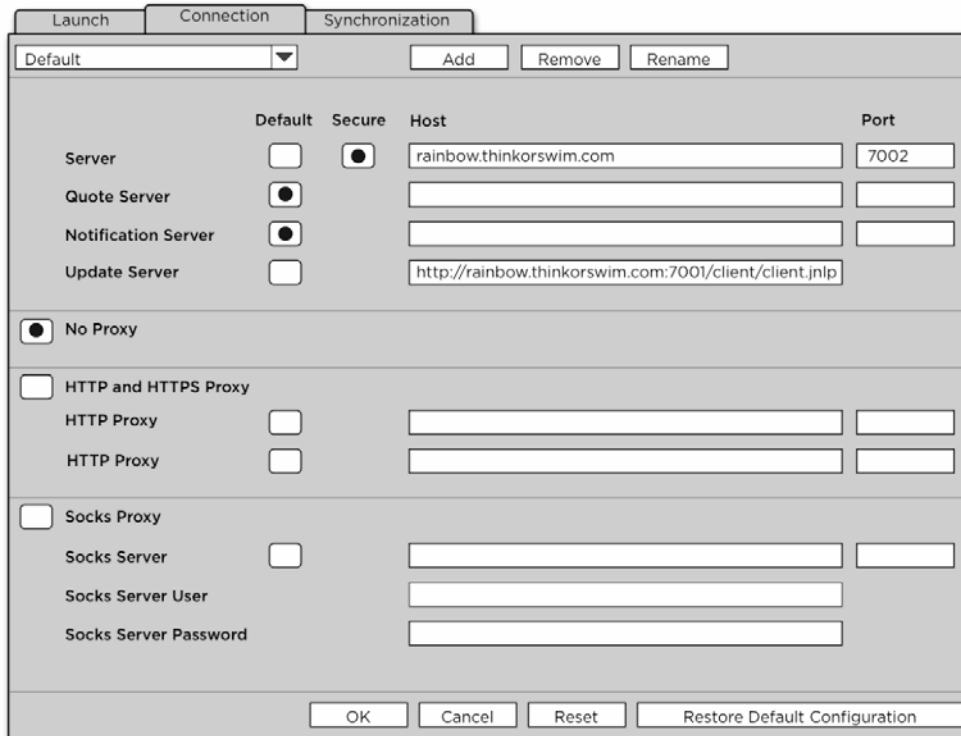
- a. The client application looks up the name 'rainbow.thinkorswim.com' and resolves it into one or more IP addresses. This is akin to looking up someone's phone number in the phone book by using his or her name. The client application must ask a DNS server to resolve the name into an IP address suitable for routing packets. In this case, and at this time rainbow.thinkorswim.com should resolve to one of these addresses: 38.103.97.141, 38.103.97.142, 38.103.97.144, and 38.103.97.152. If the client cannot get a proper resolution of the name, then no outbound connections will ever be made.

2. **Auto Update (client connects to port 7001)**
  - a. The client application will now try to connect to our servers at the resolved IP addresses and update itself. To get the list of the most current components from the servers, the client application connects to TCP port 7001 on one of the servers. The actual request is done using HTTP protocol, but targeting port 7001 instead of port 80. The response from the server will be in the form of a list that the client application saves into its installation directory and then uses it to request all the necessary .jar files. Each subsequent request for the necessary .jar files are done to port 7001 and using plain HTTP.
3. **Login and Authorization (client connects to port 7002)**
  - a. The client application components should be fully updated at this point. The client application will pass on the authentication tokens now for the first time to the servers to gain access to the user's account and position data. A connection is attempted to port 7002 using SSL.
4. **Client Account data synchronization (Continue on port 7002)**
  - a. If the authentication is successful, the client will be able to download into local memory all the private, account specific data and position data. This connection to port 7002 over SSL will remain in affect for the entire session and functions to transfer any private sensitive data to the application servers such as order placement or anything else that is considered to be sensitive information.
5. **Market Data Delivery**
  - a. If the authentication is successful the client application will attempt to connect to the servers on port 5555 and 5556. Data that travels on these two connections will not be secured. The TCP connection on port 5555 is primarily carrying Quote data. The TCP connection on port 5556 will be carrying system messaging. This data includes all general system event notifications such as security profile updates, news, and Position or Order change events.
  - b. A third set of connections will be created to support any of the historical chart features. A third party vendor – Prophet Financial, develops the historical charting component. Data to support that charting component is provided directly from Prophet's servers. The component will attempt to make a connection to port 80 at (currently) 207.33.55.212.
6. **Possible reconnection attempt if taken off-line**
  - a. In the event that the connection is lost or intentionally broken, the client will try to reconnect when it is brought back on-line by connecting first to port 7002. After negotiating authentication again the client application will initiate a connection to ports 5555 and 5556 to reestablish the quote and messaging streams. No attempt to contact the servers on port 7001 should occur again.
7. **Logoff and shutdown**
  - a. If the client application has enough time it will try to gracefully end the network connection with our servers.

### Standard Configuration

Below is visual example of the standard connection configuration in the thinkpipes application. This dialog can be reached from a button labeled "options..." in the login window. The details concerning each of these settings and their effects are on the following page. This configuration assumes a few things:

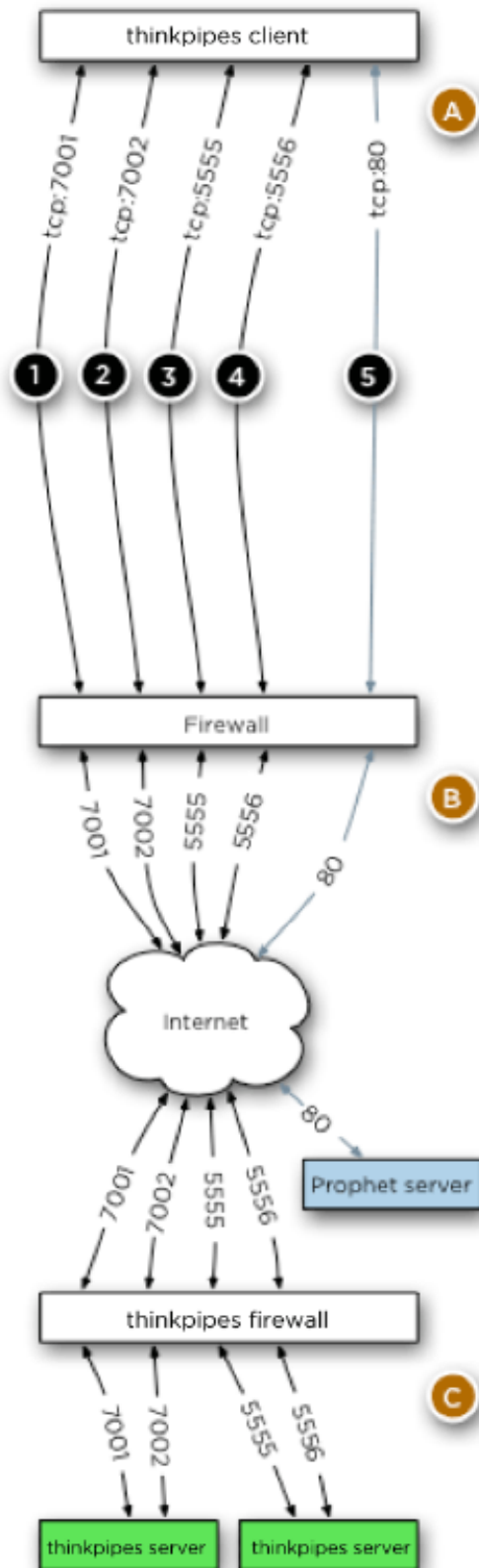
1. Client will make direct network connections to Thinkpipes servers or at most the connection is translated (NAT) through a typical firewall and no other restrictions apply.



	Default	Secure	Host	Port
Server	<input type="checkbox"/>	<input checked="" type="radio"/>	rainbow.thinkorswim.com	7002
Quote Server	<input checked="" type="radio"/>	<input type="checkbox"/>		
Notification Server	<input checked="" type="radio"/>	<input type="checkbox"/>		
Update Server	<input type="checkbox"/>	<input type="checkbox"/>	http://rainbow.thinkorswim.com:7001/client/client.jsp	

No Proxy  
 HTTP and HTTPS Proxy  
 Socks Proxy

HTTP Proxy  
 HTTP Proxy  
 Socks Server  
 Socks Server User  
 Socks Server Password



### Standard Connection Details

At the point labeled **(A)** in the diagram to the left, the client's actual IP address is most likely something private (192.168.168.2 for example).

The client application initiates all communication with the servers. As we discussed in the section entitled "Network Connection Details" the client will:

- (1) **Auto Update**
- (2) **Authenticate and Synchronize**
- (3) **Receive Market Data**
- (4) **Receive System Messaging**
- (5) **Receive Charting Data**

At the point labeled **(B)** in the diagram, the Client's Firewall has translated the internal private IP address and client's source port to a public address and port with which to connect to the thinkpipes servers. The port and IP address should remain constant for the life of the connections.

At the point labeled **(C)** in the diagram, our firewall has passed the traffic from the public IP address onto our servers and the servers should be sending data back to the client application via the translated address and port on the client's Firewall **(B)**.

The client identifies our servers by the name entered in the connection configuration dialog under "**Server**", in this case, "**rainbow.thinkorswim.com**". The "**Quote server**" and "**Notification server**" are set to "**Default**" in the configuration, which means the client will connect to the same servers to which it authenticated in Step **(2)**

**(rainbow.thinkorswim.com)** for those services as well.

Due to the load balancing mechanism implemented in the client application there is no guarantee as to which server the client is connecting for each service.

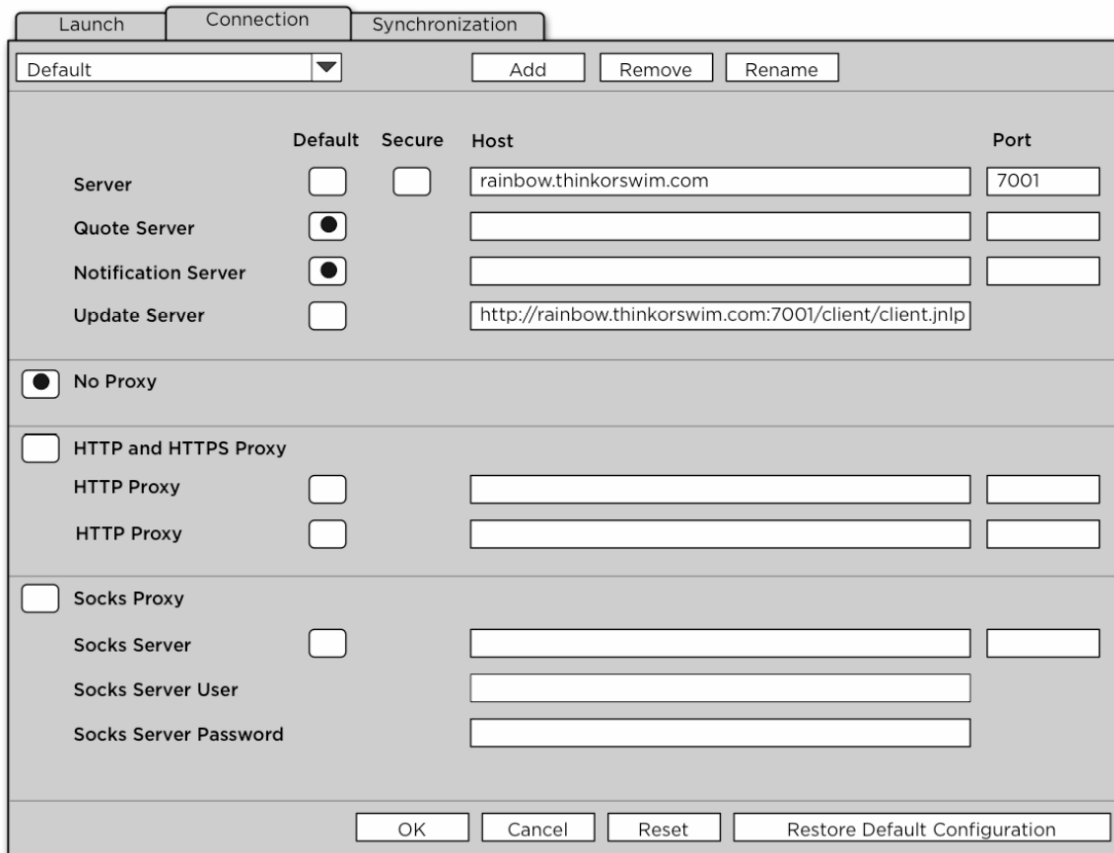
The update server entry is not set to the "**Default**" which means the client will connect to **rainbow.thinkorswim.com** on port **7001** for the auto-update process using the protocols associated with Java WebStart.

**Note** that the connection that services the charting component of thinkpipes does not end up at the thinkpipes servers. Instead, that connection is made directly to the Prophet.net server farm.

**Standard Insecure Configuration  
(for basic troubleshooting)**

Below is visual example of the standard connection configuration in the thinkpipes application. This dialog can be reached from a button labeled "options..." in the login window. The details concerning each of these settings and their effects are on the following page. This configuration assumes a few things:

1. Client will make direct network connections to Thinkpipes servers or at most the connection is translated (NAT) through a typical firewall and no other restrictions apply.
2. The connection is made over a secured line (not over the internet).



	Default	Secure	Host	Port
Server	<input type="checkbox"/>	<input type="checkbox"/>	rainbow.thinkorswim.com	7001
Quote Server	<input checked="" type="checkbox"/>			
Notification Server	<input checked="" type="checkbox"/>			
Update Server	<input type="checkbox"/>		http://rainbow.thinkorswim.com:7001/client/client.jnlp	

No Proxy

HTTP and HTTPS Proxy

HTTP Proxy

HTTP Proxy

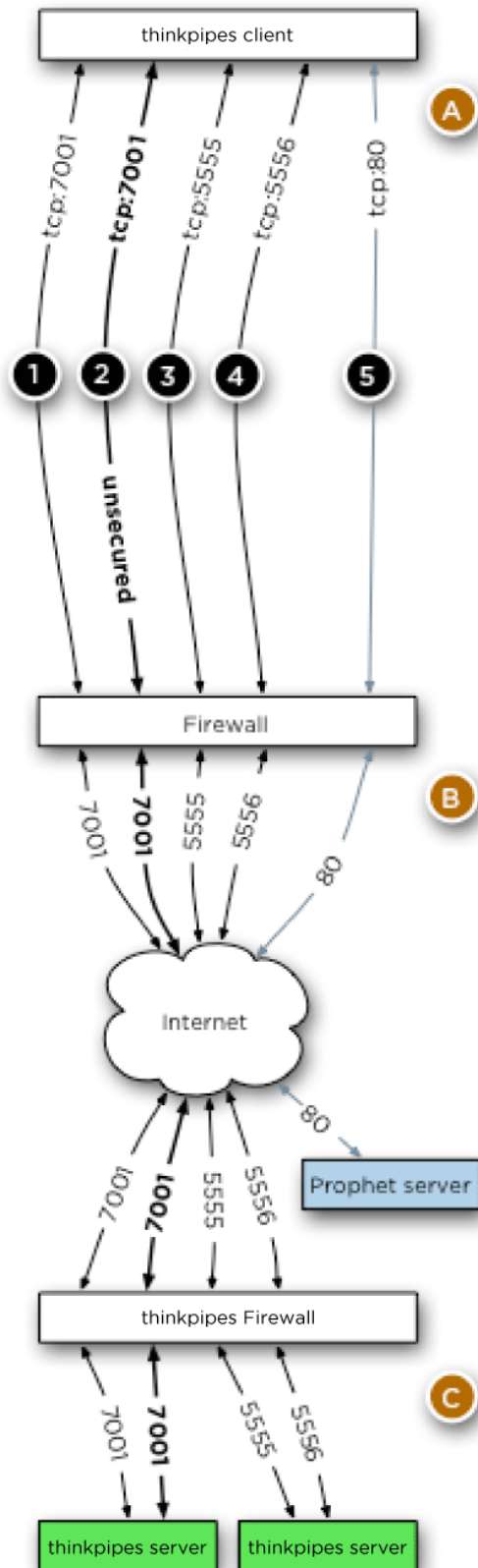
Socks Proxy

Socks Server

Socks Server User

Socks Server Password

OK Cancel Reset Restore Default Configuration



### Insecure Connection Details

This configuration is exactly the same as the default configuration except that the authentication and synchronization phases are done to port 7001 over unsecured TCP.

This configuration is only recommended for use in troubleshooting a standard configuration or for connections that are guaranteed to be secure due to their controlled access and monitoring (point-to-point links could qualify for this).

All sensitive data will be exposed over the TCP connection and anyone with access to the network traffic will be able to see the information easily.

### **SOCKs Proxy Configuration**

Some users will be forced to connect to the Thinkpipes servers through a proxy server. The configuration example below discusses the settings which you must alter in order to connect through a SOCKs proxy. This dialog can be reached from a button labeled "options..." in the login window. The details concerning each of these settings and their effects are on the following page. This configuration assumes a few things:  
(details to be TBA)