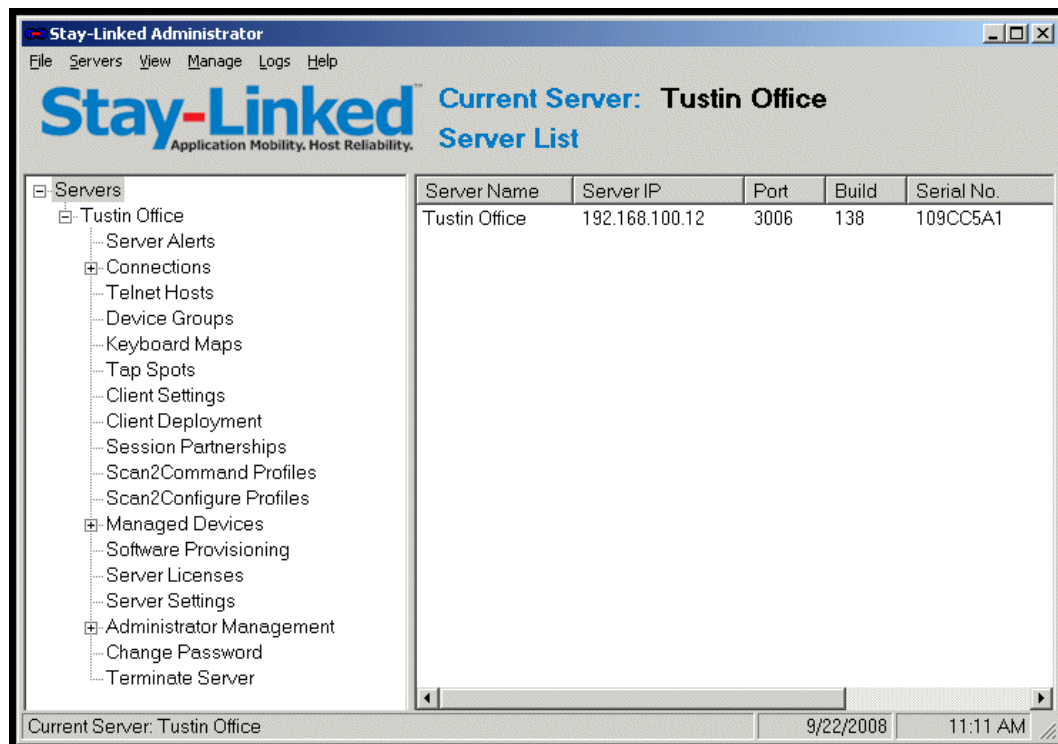


Stay-Linked™

Application Mobility. Host Reliability.

Wireless Terminal Emulation
◆
Advanced Terminal Session Management (ATSM)
◆
Device Management



Version



Stay-Linked
Secure Communications

Table of Contents



1	SECURE COMMUNICATIONS OVERVIEW.....	4
1.1	The Client2Host [®] Protocol.....	4
1.1.1	Security Compatibility.....	4
1.1.2	Additional Network Layer Security.....	5
1.1.3	Application Layer Data Encryption.....	5
1.2	Telnet, SSL and SSH Protocol Overview.....	6
1.3	The Telnet Protocol.....	6
1.4	The SSL-Telnet Protocol.....	6
1.4.1	Software Prerequisites for Stay-Linked and SSL-Telnet.....	6
1.4.2	SSL Configuration for Stay-Linked Telnet Hosts.....	6
1.4.3	Well-known or Trusted CA Server Certificates.....	7
1.4.4	Private or Third-party Server Certificates.....	7
1.4.5	Client Certificates.....	7
1.5	The SSH Protocol.....	8
1.5.1	SSH Version Support.....	8
1.5.2	SSH Transport Algorithm Support.....	8
1.5.3	SSH Authentication protocol Support.....	8
1.5.4	Keyboard-Interactive Authentication.....	8
1.5.5	Password Authentication.....	9
1.6	Security Diagram and Summary of Benefits.....	10
1.6.1	Client2Host [®] connections from the Client to the Server.....	10
1.6.2	Connections from the Stay-Linked Server to the Host Computer.....	10
1.7	Stay-Linked Administrator Secure Communications.....	11
1.7.1	Stay-Linked Administrator Communications Protocols.....	11
1.7.2	Configuring the Stay-Linked Server for a fixed TCP port range.....	11



Secure Communications Overview

This document describes the security compatibilities and capabilities of the Stay-Linked Session Management and Device Management Solution and its unique Client2Host architecture. The Stay-Linked ‘Client2Host’™ architecture overcomes all of the security challenges associated with the traditional model of running “device-side” Telnet Client software on terminal devices that communicate over the network (wireless/wired) with a host Telnet Server process and target applications. This document provides the details of how the Stay-Linked model for implementing Telnet-based device to host application communication differs from the traditional model by creating a secure “end-to-end” environment that does not feature TCP or Telnet protocols over the RF network connection at all.

In each Stay-Linked Session, there are two separate IP connections that are being managed, the Client2Host® protocol and the Telnet protocol. The Stay-Linked solution provides communications security for each of these two IP connections utilized from the device to the host application.

1.1 The Client2Host® Protocol

For the IP connection between the Stay-Linked Client and the Stay-Linked Server, the ‘Client2Host’ communications architecture is implemented. This ‘Client2Host’ communications architecture, designed from the ground up for the RF environment, utilizes a proprietary reliability protocol that is encapsulated in the UDP protocol. Because Stay-Linked uses UDP, it is able to avoid the IP traffic and connection issues that are inherent in the TCP protocol and exacerbated in a wireless environment. The Stay-Linked Reliable Protocol is quiet, only transmitting data across the network on the demand of the client. If the client device is inactive, then there will be no traffic generated on the network. The Stay-Linked Reliable Protocol implements a ‘Micro-Packet’ technology and ensures that UDP packets successfully arrive at the destination host in order. The ‘Micro-packet’ technology enables more reliable delivery of data by never transmitting packets greater than 512 bytes in length, thereby avoiding packet fragmentation and MTU issues. The ‘Micro-packet’ technology also enables Stay-Linked traffic to better utilize low-bandwidth or high-utilization network connections. Because all of the session traffic between the Stay-Linked client and the Stay-Linked Server is encapsulated in the Stay-Linked Reliable Protocol, Stay-Linked is able to avoid transmitting any TCP or TELNET protocols over the wireless network. In other words, there will be no TELNET clear-text traffic traveling across the network while running Stay-Linked.

1.1.1 Security Compatibility

The Stay-Linked ‘Client2Host’ reliable protocol is UDP-based, and thus is inherently compatible with any transport set security measures (Layers 1 – 4) that a customer might implement. For example, Stay-Linked will run well through a VPN tunnel. Moreover, even if the VPN connection is broken and then re-established, the Stay-Linked client will be able to reconnect to the existing telnet session, resuming operation at the same screen and same cursor location.

Stay-Linked is also compatible with standard RF security schemes like WEP, LEAP, PEAP, WPA, etc. This transport and RF compatibility allows customers to implement network security measures with assurance that the Stay-Linked solution will continue to operate within the new architecture. Additionally, Stay-Linked is compatible with various authentication methods such as Kerberos, RADIUS, etc. Stay-Linked sessions themselves are authenticated by the Telnet Host’s login process.

1.1.2 Additional Network Layer Security

Stay-Linked supports some traditional network security configurations by implementing a firewall-friendly communication architecture. The Stay-Linked Server supports Static-NAT configurations and allows for configuration of Port Restriction rules in order to be compatible with any port filtering switch or firewall. In the case of port filtering, since Stay-Linked is UDP-based and transmits no TCP or TELNET over the network, you can disable all TCP traffic through the switch or firewall, including ports usually supporting the TELNET protocol. Further, you can restrict open UDP ports to only those required to connect your licensed number of Stay-Linked devices.

Following are a description of the UDP ports used by the Stay-Linked Client2Host protocol:

Ports used for inbound communications from the Device (source) to the Server (destination)

Inbound packets are typically filtered at the firewall. These are the UDP ports that will be used by the Stay-Linked Server.

UDP Port 3006 - This is the port where the Stay-Linked Server is listening for connection requests.

UDP Port Range - These are the ports that will be dedicated to each device connection to the server. This is a user-defined port range that is configured in the Stay-Linked Administrator->Server Settings->Firewall Settings. You will need enough ports opened up in the firewall so that each device session will have an available port. Usually, this is based upon how many connections are licensed.

Ports used for outbound communications from the Server (source) to the Device (destination)

Usually, outbound packets are not filtered at the firewall. But, if you need to open outbound ports, these are the UDP ports used by the Stay-Linked Client.

UDP Port 3771 - This is the port used by the Stay-Linked Client to connect to the Stay-Linked Server.

UDP Port 3772 - This is the port used by the Stay-Linked Client to query session status during any communication interruptions.

UDP Port 3773 - This is the port used by the Stay-Linked Client to listen for any TFTP transfers from the Stay-Linked Administrator (legacy file transfer mechanism).

UDP Port 3774 - This is the port used by the Stay-Linked Client to connect to the Stay-Linked Server for the second session when running 'dual sessions'.

1.1.3 Application Layer Data Encryption

Stay-Linked provides two different data encryption options for securing the data that is transmitted between the Stay-Linked Client on the device and the Stay-Linked Server on the host. The data encryption feature applies to all connections

Level 1 Encryption – This encryption method utilizes a native, proprietary, dynamic-symmetric 64-bit key, stream-cipher symmetric encryption algorithm to encode the cargo of the Stay-Linked 'Client2Host' reliable protocol packets. This encryption technology is designed to be compatible with older, legacy devices that are not powerful enough to support the high-end, public encryption algorithms. This encryption option is available for all Stay-Linked clients on all platforms.

Blowfish Encryption – This encryption method provides a very high level of data security for the Stay-Linked packets that are transmitted between the Stay-Linked Thin-Client and the Stay-Linked Server. Stay-Linked Blowfish encryption utilizes the Blowfish/ECB/PKCS5Padding cipher. You may also specify key rotation with up to four keys defined and you can specify a key rotation interval. This encryption option is available only for PPC/WM/CE/CE.Net devices that are running a Stay-Linked Client of at least Version 9.1.0. Blowfish encryption is not available for any DOS devices.

1.2 Telnet, SSL and SSH Protocol Overview

At the heart of the Stay-Linked Host-based Terminal Emulation architecture is the Stay-Linked Server which runs directly on the same host platform as the telnet server and host applications. This host-based architecture provides an inherent level of security by isolating the telnet connection and communication within the host platform. In this host-based architecture, there are no telnet protocol packets transmitted across the network and no clear-text data exposed to the network. Additional levels of terminal emulation security are also supported by the Stay-Linked solution. In those cases where SOX or PCI compliance are required, the Stay-Linked solution supports both SSH and SSL-Telnet terminal emulation connections to the host system. This section will describe the Stay-Linked support for Telnet, SSL-Telnet and SSH connections to host systems.

1.3 The Telnet Protocol

Normal telnet connections feature the telnet protocol transmitted from the telnet server to and from the telnet client. The telnet client will typically connect to the telnet server on port 23 which is the official Telnet port. The data carried by the telnet protocol is not encrypted and contains information in clear-text. This protocol is TCP-based and provides the telnet connection between the Stay-Linked Server, running on the host, and the Telnet server which should be also running on the same machine. When the Stay-Linked Server and Telnet server are running on the same machine, the Telnet protocol is transported on the 'local loop back' connection which is internal to the machine and is inherently reliable. In some cases, the Telnet server may not be running on the same hardware as the Stay-Linked Server. Under these conditions, the Telnet protocol is transported over the regular network topology and is naturally more exposed to observation, failure or interruption.

1.4 The SSL-Telnet Protocol

In order to secure the data carried by the telnet protocol and also to secure the connection to a telnet server, the SSL (Secure Socket Layer) protocol may be employed. With SSL-Telnet, the telnet client will connect to the telnet server on port 992 which is the official SSL-Telnet port. In order to run SSL-Telnet connections, both the telnet server and the telnet client must support the SSL protocol. Usually, the telnet server and telnet client (Stay-Linked) will provide options for configuring SSL support.

1.4.1 Software Prerequisites for Stay-Linked and SSL-Telnet

In order to run SSL-Telnet connections with Stay-Linked, there are some minimum software requirements. If you install Stay-Linked Server Version 8 Build 134 or later, then SSL support is already included. If you are upgrading from a previous version of Stay-Linked, then you will need to make some minor changes to your server configuration in order to enable SSL support. Stay-Linked SSL Support will require the addition of two Java JAR files to the '..\stay-linked\lib' folder, 'sls.jar' and 'slt.jar'. Additionally, these two JAR files must be added to the 'classpath' in the Stay-Linked Server Startup Script named 'strserver.sh'. Also, there is an updated version of the 'hostprops.xml' file to be used by the Stay-Linked Administrator for configuring Telnet Hosts to use SSL. In order to upgrade your Stay-Linked software for SSL-Telnet support, please contact Stay-Linked Technical Support for assistance.

1.4.2 SSL Configuration for Stay-Linked Telnet Hosts

Once your Stay-Linked Server is configured to provide SSL-Telnet support, it is very simple to configure an SSL-Telnet connection for your wireless devices. The Stay-Linked Administrator will be used to configure a 'Telnet Host Entry' for SSL connections. There are a number of new

'Emulation Properties' that can be added to a 'Telnet Host Entry' to enable the various levels of SSL support that Stay-Linked provides. Here is a listing of the SSL-related Emulation Properties:

SSL Session - To request an encrypted session, set this property to True. This is the minimum configuration required to run SSL-Telnet sessions.

SSL Client Certificate Provided - Determines whether the client has a certificate.

SSL Client Certificate URL - URL (Path) of the client certificate.

SSL Client Certificate Password - Password of the client certificate.

1.4.3 Well-known or Trusted CA Server Certificates

The simplest implementation of SSL-Telnet with Stay-Linked is to use a Well-known or Trusted CA Server Certificate for your telnet server application. If you configure your telnet server to use a Well-known or Trusted CA Server Certificate, then the only configuration option required for the Stay-Linked Telnet Host Entry is to add the '**SSL Session**' Emulation Property and set the value to '**True**'.

1.4.4 Private or Third-party Server Certificates

You can also use a Private or Third-party Server Certificate with Stay-Linked, but there is some additional configuration required in addition to setting the '**SSL Session**' Emulation Property. First, the Private Server Certificate must be exported into a '.p12' (PKCS12, Personal Information Exchange Syntax Standard) file format. The file must be named 'CustomizedCAs.p12'. The password for this file must be set to 'hod'. Second, the file must be placed in the Stay-Linked Server main folder, '..\stay-linked'. With this file properly created and located, the Stay-Linked Server will be able to operate using your Private or Third-party Server Certificate.

1.4.5 Client Certificates

Client authentication is similar to server authentication except that the telnet server requests a certificate from the client to verify that the client is who it claims to be. The certificate must be an X.509 certificate and signed by a certificate authority (CA) trusted by the server. You can only use client authentication when a server requests a certificate from a client. Not all servers support client authentication. Client Certificates can be obtained from a Certificate Authority or can be a Self-signed Certificate. The Client Certificate should be exported to a password-protected PKCS12 file. The public portion of a self-signed client certificate should be added to the telnet server's trusted list. The Client Certificate must be placed in the Stay-Linked Server main folder, '..\stay-linked', on the server. Now, you must add three additional 'Emulation Properties' to the 'Telnet Host Entry' using the Stay-Linked Administrator:

SSL Client Certificate Provided – Set this value to True to indicate that the client has a certificate to provide to the server.

SSL Client Certificate URL – Set this value to the file name of the client certificate.

SSL Client Certificate Password – Set this value to the password of the client certificate

1.5 The SSH Protocol

For VT emulation, an alternative to the telnet protocol is the SSH (Secure Shell) protocol. The Secure Shell (SSH) is a set of protocols for implementing secure sessions over a non-secure network (such as a standard TCP/IP network). In order to use SSH, you must set up SSH server software on the host. Security features include the following:

- Secure login
- Strong authentication of server and client
- Several user authentication methods
- Encrypted sessions

1.5.1 SSH Version Support

The following table describes the SSH Version support currently implemented in Stay-Linked:

Version of SSH	Supported by Stay-Linked
SSH Version 2.0	Yes (subset)
SSH Version 1.5	Yes (subset)
SSH Version 1.3	No

1.5.2 SSH Transport Algorithm Support

The following table describes the SSH Transport Protocol algorithms currently supported by Stay-Linked:

Category	Algorithm supported
Compression:	None
Encryption ¹ :	3des-cbc aes128-cbc
Data Integrity:	hmac-sha1
Key Exchange:	diffie-hellman-group1-sha1
Public Key:	ssh-dss (same as DSA), ssh-rsa

¹ Stay-Linked always gives priority to 3des-cbc over aes128-cbc. If you want to use aes128-cbc, 3des-cbc needs to be disabled on the server side.

1.5.3 SSH Authentication protocol Support

For the SSH Authentication protocol, Stay-Linked supports the following:

- Keyboard-Interactive (SSH Version 2)
- Password (SSH Version 2 and SSH Version 1.5)

1.5.4 Keyboard-Interactive Authentication

Configuring your SSH Server for keyboard-interactive authentication

The server configuration for keyboard-interactive authentication differs depending on the vendor or source of the SSH support. Refer to the documentation for your SSH server software for information on how to configure the SSH server for the keyboard-interactive authentication method.

Configuring the Stay-Linked Server for keyboard-interactive authentication

You do not need to configure Stay-Linked for keyboard-interactive authentication. The Stay-Linked SSH client will look for whether or not keyboard-interactive authentication is configured on the server. If it is configured on the server, then Stay-Linked will prompt the user for keyboard input. Stay-Linked will display a logon prompt according to the data received from the host. This logon

prompt allows you to input responses from the keyboard. If your server is configured for keyboard-interactive authentication, then the Stay-Linked 'Startup Scripts' will not be available for automated logon because manual keyboard entry of the user and password is required.

1.5.5 Password Authentication

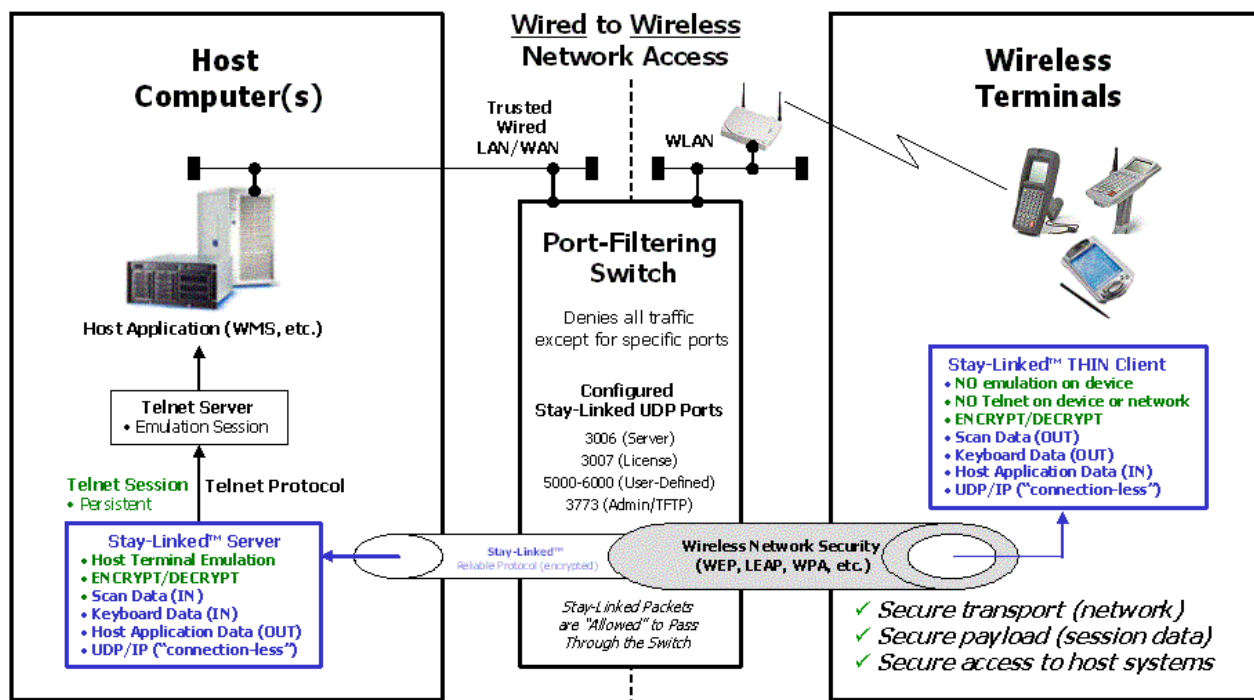
Configuring your SSH Server for password authentication

The server configuration for password authentication differs depending on the vendor or source of the SSH support. Refer to the documentation for your SSH server software for information on how to configure the SSH server for the password authentication method.

Configuring the Stay-Linked Server for password authentication

You do not need to configure Stay-Linked for password authentication. If the SSH Server is not configured for keyboard-interactive authentication, then the Stay-Linked SSH client will display a login prompt for a user id and password. The user can type-in their user id and password or the Stay-Linked 'Startup Scripts' can be implemented for automated logon.

1.6 Security Diagram and Summary of Benefits



Copyright 2004 – eBusiness Solution Pros, Inc.

1.6.1 Client2Host® connections from the Client to the Server

- Utilizes the connection-less UDP protocol for WLAN-friendly communications
- Client2Host reliability protocol is 'quiet', reducing network utilization
- Implements 'Micro-packet' technology for reliable data transmission
- No 'Clear Text' is ever transmitted across the network
- No TCP protocol is ever transmitted across the network
- Compatible with transport set (Layer 1–4) security protocols
- Compatible with wireless network security protocols
- Compatible with standard authentication protocols
- Firewall-compatible communication architecture
- Allows for the filtering of the common, high-risk TCP protocols
- Supports proprietary 'Level 1' encryption for Client to Server communications
- Supports highly secure Blowfish encryption for Client to Server communications

1.6.2 Connections from the Stay-Linked Server to the Host Computer

- Telnet protocol is supported and typically contained completely within the host computer
- Secure SSL-Telnet protocol is also supported for communications with the host computer
- Secure SSH protocol is also supported for communications with the host computer

1.7 Stay-Linked Administrator Secure Communications

The Stay-Linked Administrator console application runs on a Windows-based computer and communicates with the Stay-Linked Server for the purposes of configuration and for session management. These tasks will typically be performed from within the same network as the Stay-Linked Server and so, no special network communications configuration is required.

In some cases it may be desired to use the Stay-Linked Administrator to manage the Stay-Linked Server through a firewall. In this case, a special Stay-Linked Server configuration and special firewall access rules will need to be coordinated so that the Stay-Linked Administrator management traffic can navigate through the firewall.

1.7.1 Stay-Linked Administrator Communications Protocols

The Stay-Linked Administrator uses two different protocols to communicate with and manage the Stay-Linked Server. The Stay-Linked Administrator sends commands to UDP Port 3006 and received responses from UDP Port 3006. Some of these commands will instruct the Stay-Linked Server to provide additional data to the Administrator. This additional data will be delivered through a TCP socket connection that is opened on the server. The Stay-Linked Administrator will attempt to connect to the TCP socket on the server in order to send or receive additional data to or from the Stay-Linked Server. In order for these communications to be able to navigate through a firewall, you will need to open UDP Port 3006 for the Administrator commands, and also a fixed range of TCP Ports for the TCP socket connections.

1.7.2 Configuring the Stay-Linked Server for a fixed TCP port range

In order to have the Stay-Linked Server open TCP ports only in a specific range that will match your Firewall rules, you must configure the Stay-Linked Server for that specific TCP port range. You will accomplish this configuration by manually modifying one of the server configuration files. The specific Stay-Linked Server configuration file that you will modify is located on the server machine:

```
../stay-linked/config/espadmin.xml
```

In this file, you will add an additional XML node just above the last line of the file, like this:

```
<espadmin>
  ...
  ...
  ...
  <adminfirewall lowport="5000" highport="5009"/>
</espadmin>
```

You will want to specify at least 10 available ports in the range that you select. This should be enough ports to support multiple Stay-Linked Administrator consoles managing the server concurrently. This TCP port range should be opened in your firewall access control list in addition to UDP Port 3006.

Once you have saved the changes to this file, there is no additional action required on the server. These new settings will take effect the next time the Stay-Linked Administrator console connects to the Stay-Linked Server.

For further information or technical assistance, please contact your Stay-Linked Certified Partner or you may contact us directly...



This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).