## Introduction

Resilio Connect uses cryptographic security that is built on industry standards. The implementation leverages OpenSSL cryptographic libraries that are used on Windows, MAC and Linux, as well as OS provided cryptographic APIs (Windows and OSX).

**The Resilio Connect security model consists of:**

- Mutual authentication and authorization of clients and Management Console
- Generation of one-time session encryption keys between clients
- Data in transit encryption
- Data integrity validation

## Key features

- Works inside your private infrastructure
- Uses industry standard crypto algorithms: AES 128-bit (AES 256-bit), SHA1 (SHA2)
- Incorporates SRP for session establishment and forward secrecy
- Data integrity is based on the SHA1 and ED25519 signature algorithm
- Endpoint authentication and authorization over TLS

## Cross-Platform Support

**Management Console:**

- Windows XP (SP2) or later (32/64-bit)
- Linux x64 & x64 (glibc 2.3)

**Resilio Connect's Clients:**

- Mac OS X 10.8 Lion or later
- Windows XP (SP2) or later (32/64-bit)
- Linux i386 & i386 (glibc 2.3)
- Linux x64 & x64 (glibc 2.3)

## Session Encryption

The Resilio Connect's client receive a 160 bit (20 bytes) private folder key from the Management Console. The private key indicates that the client has either read-write (RW key) or read-only (RO key) access to a folder. The client must have a folder private key before it can initiate a session with other clients.

The Resilio Connect's client uses SRP with the folder private key (RW key or RO key) to do mutual authentication of clients and to generate a session key for traffic encryption. The transfer key is unique to each client, folder and session. The use of SRP for session key generation ensures perfect forward secrecy.

## Data Integrity

The Resilio Connect client that has the RW key can change the content of the folder. It detects when the content of the file is changed, then it splits the file into blocks (32KB or more) and calculates the hash (SHA1 or SHA2) of each block as well as a hash of all of the files blocks. This information is used to verify that the received block has arrived without corruption. The receiver can also verify that file is fully delivered by calculating the hash of all of the files blocks. It can also retransmit only damaged blocks, without the need to resend the entire file.

Information about the directory is passed as a part of folder meta information. Every piece of metadata is signed and signature is verified by all clients during synchronization.

RO key is derived from RW key using ED25519 key generation algorithm. RW key is used to sign and to verify meta information, RO key can be used only to verify meta information.

Resilio Connect clients that have the RW key can modify folder meta information (add/change files) and sign changes. This guarantees that changes to the folder can only be made by clients that have the RW key.

## Data Is Encrypted In Transit

The Resilio Connect client uses SRP to do mutual authentication of other clients and to generate 128 bit session keys for data transfer.
The Resilio Connect client uses AES 128-bit in CTR mode to encrypt all communication between clients. This includes exchange of folder meta information, actual file data and control messages.

The Resilio Connect client uses persistent connections over TCP or UDP protocols to transfer the encrypted packets.
The session keys are discarded when the connection between clients are terminated.

## Client Authentication

**Resilio Connect's clients use TLS to communicate with the console.** This way all communication is encrypted by using industry standard encryption.

The Resilio Connect clients must be authenticated against the Management console to connect and communicate with it. This is achieved through bootstrapping a Resilio Connect's client with a bootstrap token (20 bytes) with limited time to live. The bootstrap token is generated by the console and the console can change or revoke it at any time. Each new Resilio Connect's client must provide a valid bootstrap token to establish a connection to the console. Bootstrap token is supplied to the Resilio Connect's client through client configuration file during installation.

Once the client is authenticated with a valid bootstrap token, the console issues a unique client token (20 bytes) that the client needs to provide during connection to the console. The client validates the console authenticity by checking the console certificate fingerprint during TLS handshake. The console certificate fingerprint is a part of the client configuration.

**Every Resilio Connect's client mutually authenticate before any data is being sent.** Data transfers only happen between clients that were authorized by the console to do so.

## Networking

**The Resilio Connect's client does the following network activities:**

- Communication between clients over TCP and UDP
- Communication with tracker over TCP and UDP
- Search for local peers using multicast UDP packets
- Communication with the Management Console over TCP

**Client listening sockets:**

- TCP socket for incoming TCP connections on a random port in the range of 10000-65536 or other value set in configuration.It is bound to all network interfaces..
- UDP socket for incoming and outgoing UDP communication bound to all network interfaces.  It uses the same port as the TCP socket
- UDP socket for every network interface bound to local scope multicast address 239.192.0.0 on port 3838 to listen for LAN discovery packets

**Console listening sockets:**

- TCP socket for management UI over https (default is 8443)
- TCP socket for managing Resilio Connect's client (default is 8444)
- TCP socket for getting audit and debug logs from clients (default 8445)

**For communication with the tracker the Resilio Connect's clients uses TCP and UDP.** If both connections succeed, Resilio Connect prefers UDP.  UDP connection allows the tracker to see the actual outgoing UDP port used for communication. This port is reported to other peers and used for NAT traversal. Resilio Connect's client connect to tracker on start and keep persistent connection. When new client comes online, tracker sends notification to already connected clients with address of new client.

To search the LAN for other clients with the same folder, Resilio Connect sends UDP packets to multicast address 239.192.0.0:3838. If there are other clients on the LAN with the same folder, they reply to the sender of multicast packets.

Every Resilio Connect's client has a list of clients having the same folder. Resilio Connect keeps persistent connections to every client from this list. When any change occurs on any client, it notifies all other connected clients and it triggers synchronization. For communication between clients Resilio Connect uses both TCP and UDP. Resilio Connect prefers TCP for LAN connections and UDP for WAN connections. Using custom TCP-like protocol over UDP allows to adjust congestion control according to network conditions. Also it allows to establish connection to other clients behind NAT.

For communication with the Management Console Resilio Connect's clients use TLS 1.0 - 1.2 (depends on what is supported by client) over TCP. Every Resilio Connect's client keeps persistent connection to management port (default is 8444) which is used to get configuration and report status. Also, clients may open connection on demand to log uploading port (default 8445) to send audit or debug logs to console.

## Private Infrastructure

Resilio Connect provides the ability to run completely within private infrastructure. It doesn't require any external web services or other resources to deploy policies and transfer data between clients. Resilio Connect's client use multicast to find other clients that have the same folder. In addition a private tracker is deployed on-prem to enable client discovery over networks where multicast is blocked or not available.

The private tracker keeps information about all the clients that share the same folder. The Resilio Connect's client reports to the tracker the list of folders that it has and receives list of other peers that has the same folders. This way peers could find each other and establish connection without using multicast.

## Client Security

The Resilio Connect's client is a single binary that has no dependencies on external libraries and frameworks.This significantly simplifies client installation and gives easy upgrade path for the endpoint systems.

The Resilio Connect's client uses limited number of ports for all communications with other machines and Management Console. This makes firewall rules configuration extremely easy for all devices inside your network.

The Resilio Connect's client doesn't require any administrative privileges to run. It could run in a sandboxed environment or under a user with limited permissions.

## Security Review

The Resilio Connect security design and implementation was reviewed by 3rd party security auditor.