

# How FireCast keeps your data secure

---

WireSpring takes security very seriously, and this document provides an overview of our security protocol.

## Security in FireCast OS and ClientCenter

The FireCast platform is based on secure, open standards, and is vigilantly maintained to be up-to-date and protected against the latest security vulnerabilities. Because of its unique Linux architecture, FireCast OS is inherently immune to the vast majority of viruses and worms.

FireCast OS is based upon the proven and trusted Linux platform. WireSpring has removed many unnecessary utilities from the core operating system in an effort to reduce complexity and improve security. FireCast OS does not accept incoming connections on any ports, and connects to ClientCenter using only a single outbound HTTP connection over port 80 (or HTTPS over port 443), making it very firewall-friendly. There are no FTP, SSH, Telnet or other servers running, making the system extremely difficult to remotely exploit. Additionally, the core FireCast libraries are encrypted for added security.

All communication between the kiosk/signage units and ClientCenter is encrypted, using a 128-bit Blowfish algorithm and a 448-bit symmetric key. Authentication between the units and the ClientCenter server uses shared secret exchange to generate temporary 128-bit, one-way MD5 digest keys. Session keys are good for one hour, after which point they are expired and the units must re-authenticate to generate a new, valid key. On networks that support it, FireCast OS-powered units can also wrap all kiosk-server communications in a 128-bit Secure Sockets Layer (SSL) connection for additional security.

The ClientCenter remote management service utilizes a 128-bit Secure Sockets Layer (SSL) connection, ensuring that your schedules, reports, and other assets are protected using the same methods as modern online banking sites.

All communications are logged, and WireSpring's staff continuously monitors the logs and other server properties for suspicious or anomalous behavior.

## Security at our datacenters

WireSpring's servers are collocated in secure datacenters operated by Rackspace Managed Hosting, the world's fastest-growing managed hosting company. WireSpring and Rackspace manage server and infrastructure security on three different levels: physical security, network security and account security.

### Physical Security

Data center access is restricted to Rackspace's level three technicians. 24/7 security guards, Biometric palm scanners and a military-grade passcard system are in use to ensure that no unauthorized persons can access our servers.

## Network Security

Edge-of-network and switch-level security devices handle packet-level hacker attacks like ping and SYN floods, core router attacks, etc.

WireSpring's servers run a custom-hardened version of RedHat Advanced Server, following the SELinux guidelines published by the National Security Agency (NSA). By default, our ClientCenter servers will only listen to inbound connections on port 443 (SSL), and only allow HTTPS connections for authorized ClientCenter users.

Remote access to the ClientCenter servers is disabled by default. In the event that WireSpring needs to start a remote access session with a ClientCenter server, the connection must be initiated from WireSpring's headquarters office in Ft. Lauderdale, FL, USA and established via a secure 128-bit SSH tunnel.

## Account Security

Rackspace maintains a list of Authorized Contacts and a Secret Question, both of which are supplied by WireSpring. Rackspace will not provide any information regarding our servers or account or any technical support to anyone other than WireSpring's authorized contacts - and only then if they know the answer to our Secret Question.

## We're here to help

If you have any questions, feel free to contact us at +1 954-548-3300 or email [support@wirespring.com](mailto:support@wirespring.com). For additional contact options, including live chat, please visit [www.wirespring.com](http://www.wirespring.com) and click on **Support**.