How the FileWave Client Communicates

Enrollment

At this point you have either installed the FileWave client onto a computer, or the computer has gotten the client from DEP.

1. Client uses server field to make a connection to the server
2. Client downloads CA info from server
3. Client generates client ID
4. Client generates CSR (Certificate Signing Request), and saves key
5. Client sends client ID, name, CSR to server
   1. Client polls server (on tickle/heartbeat interval) until certificate is ready
   2. Server sends the client certificate
   3. Client verifies the certificate against the local key and the CA it downloaded
6. Client is added (and certificate is created)
   1. (Auto-add) Client is automatically approved and goes into the selected automatic add folder
   2. (upgraded client) Client is automatically approved, and continues normal operation
   3. (Vetting) Client sits in the new client UI (Client view → New client → Desktop Clients) till approved

If the client is older than 13.1 it will not connect if compatibility mode is disabled (see: What is Compatibility Mode?)

Daily Communication

FileWave server is updating manifests both when the model updates and when calculating smart group updates (see inventory preferences for time variable), so when a check-in happens the model may not have changed, but there may be new manifests for the client nonetheless.

macOS and Windows

Tickle

1. Checkin (AKA Tickle) interval (Changeable with Superprefs, see: Creating a Superprefs Fileset) has passed
2. Client reaches out to server
3. Server verifies certificate
4. Server responds with model number
   1. If the model number on server matches client, then no model manifest is download
   2. If the model number does not match the server, client downloads manifest from server
5. Client processes manifest(s)
6. Client reaches out to either server or booster and downloads fileset(s)
   1. (if booster) Client reaches out to booster requesting fileset ID
   2. Client verifies the boosters certificate is valid against the same CA (Certificate Authority - AKA your FileWave server) that the client uses.
   3. Booster checks the client certificate against the CA it downloaded from the server and the CRL (Certificate Revocation List - Certs that have been made invalid)
   4. If the certificates (booster or client) are not valid, not signed by the CA or part of the CRL, the TLS handshake fails and the connection is dropped (with no data being transferred), the failure is reported to the server.

Verify

1. Additional check-in, by default every 24hrs. ('File Check Interval', changeable with Superprefs, see: Creating a Superprefs Fileset)
2. After Server/Client certificate confirmation, the client confirms status of all Filesets and Inventory and 'heals' any files as configured within Filesets.

> Verification also occurs when the client service commences (e.g. reboot) and can be triggered manually through Client Monitor/Info or command line on the device.

Android EMM

Review the topology on Default TCP and UDP Port Usage for a visual on how things connect.

At Model Update

1. A single manifest is created from all Android Policy Filesets (AKA Policy Fragments) associated to the device
2. Any App associated to the device (including app permissions) are added to the manifest
3. Manifest is sent to AMAPI (Android Management API)
4. EMM Android device reaches out to Google

Rest of the time

1. FileWave reaches out to Google every 5min to check for new devices and verify activity
2. If there are smart-groups update, a new manifest would be created and sent to AMAPI (see Android EMM Known Issues KB)

iOS

Review the topology on Default TCP and UDP Port Usage for a visual on how things connect.

At Model Update

1. At model update a push notification is send to the device asking it to verify
2. The iOS device connects with the FileWave server
3. The new changes in the manifest are sent

Rest of the time

1. Next inventory interval is hit
   Default is 24hrs. Configured in: Inventory preferences Meaning the last time we have talked to the device. So if you updated the model at 4pm on Friday, then 4pm Saturday would be the next interval
2. FileWave sends out a Push to Apple asking the device to talk to FileWave MDM server
3. The iOS device connects with the FileWave server
4. The new changes in the manifest are sent

Revision #10
Created 21 June 2023 19:59:15 by Josh Levitsky
Updated 25 March 2025 15:17:59 by Josh Levitsky