1. The host key is used to sign the Diffie-Hellman parameters. It is used during the key exchange; RSA is a signature algorithm as well as an encryption algorithm. From what I can tell, the client key (in authorized\_keys) is not used in key exchange at all; it's only used after key exchange to sign a particular message and prove the client has the private key (one side of DH parameters being signed is enough to prevent a MITM, because the attacker can't impersonate both client and server; it's easier to make the server always have to have a keypair than make the client always have to have a keypair).
2. The way SSH uses DH is as an ephemeral algorithm: DH parameters are generated for individual sessions, and are destroyed as soon as they're no longer needed. The only thing the long-lasting keypair is used for is authentication. This gives forward secrecy: stealing the private key doesn't let you decrypt old sessions.

If RSA was used to encrypt the session key, then someone who recorded that session, buys the SSH server years later after it's been decommissioned, and obtains the private key off its hard drive can use that to decrypt the session key, and use that to read the entire communication. If RSA is used to sign DH parameters, then the only way to exploit a stolen private key is in a man-in-the-middle attack, and that can be foiled by the server operator changing the key and telling all his users about the changed keys. With ephemeral key exchange, nothing needed to decrypt a recorded session is stored any longer than it has to be.

This is also a configuration being used increasingly often for TLS sessions (there, where certificates have expiration dates, a private key doesn't even strictly have to be kept secret after its expiration because it shouldn't be possible to use it in a man-in-the-middle attack; it is, because users will ignore expired certificates, but you can see why ephemeral encryption is nice).