Changing the Host Key of Tectia Server

Tectia Server version 6.4.19 has a feature to help with changing the host keys on the client side. To use it, you can configure host key rotation on the server-side. This will allow clients that authenticate the server with the old host key to save the new host key after successful user authentication and delete the old one once the old key is removed on the server-side, for example after 3 months. This feature requires a Tectia client version 6.4.19 that has Host Key Policy Rotation enabled (by default enabled when connecting to Tectia servers only) or a OpenSSH client version 6.8 or above that has UpdateHostKeys enabled.

Quick Comparison:

- Manual Change
    - Change host key without advertising it first. All secure shell clients that have previously connected and saved the old host key to known hosts fail to connect or prompt a host key changed warning.

- Automatic Rotation
    - Time-based key generation, advertising and rotation that changes the host key
    - hostkey (current advertised and used as server identity)
    - hostkey.next (new advertised)
    - hostkey.old (previous hostkey that has been removed from configuration)
    - Same algorithm and key size as the current hostkey
    - Must not be enabled for Tectia Server cluster nodes
    - Server_hostkey_rotation_started and Server_hostkey_rotation audit messages

- Manual Rotation
    - Administrator controls new key generation, advertising and changing the host key
    - Host key algorithm or key size can be different for new key
    - If Tectia Server is part of a cluster the new host key has to be shared on all nodes and advertising needs to be enabled and disabled for all keys on a node. Advertising must not be enabled on other node unless it has the same current and new host keys. Also, advertising must be disabled for all keys on a node before new key is taken into active use and advertising can only be enabled again once all nodes have taken the same new key into active use.

Host key Algorithm in Manual Host Key Rotation

The hostkey algorithm should be decided based on security policy. The new host key with different algorithm could be taken into active use faster in the environment than using the same algorithm as the current host key but it may cause unexpected key exchange failures if the different algorithms are not allowed in configuration.

While Tectia Server can have multiple identities, the client and server can only agree on one hostkey algorithm during key exchange in secure shell protocol. The negotiated algorithm depends on what the server offers and what the client supports and prefers in its configuration and what type host key(s), if any, it has saved to known hosts for the server.

For example if Tectia Server has currently RSA hostkey and new host key is generated with different algorithm for example ECDSA or ED25519 both can be enabled simultaneously as current hostkey identities. Clients that connect the first time may already use the new hostkey but clients that prefer or only support RSA or clients that have connected before continue to use the RSA hostkey as long as the server has it enabled and offers it in key exchange.

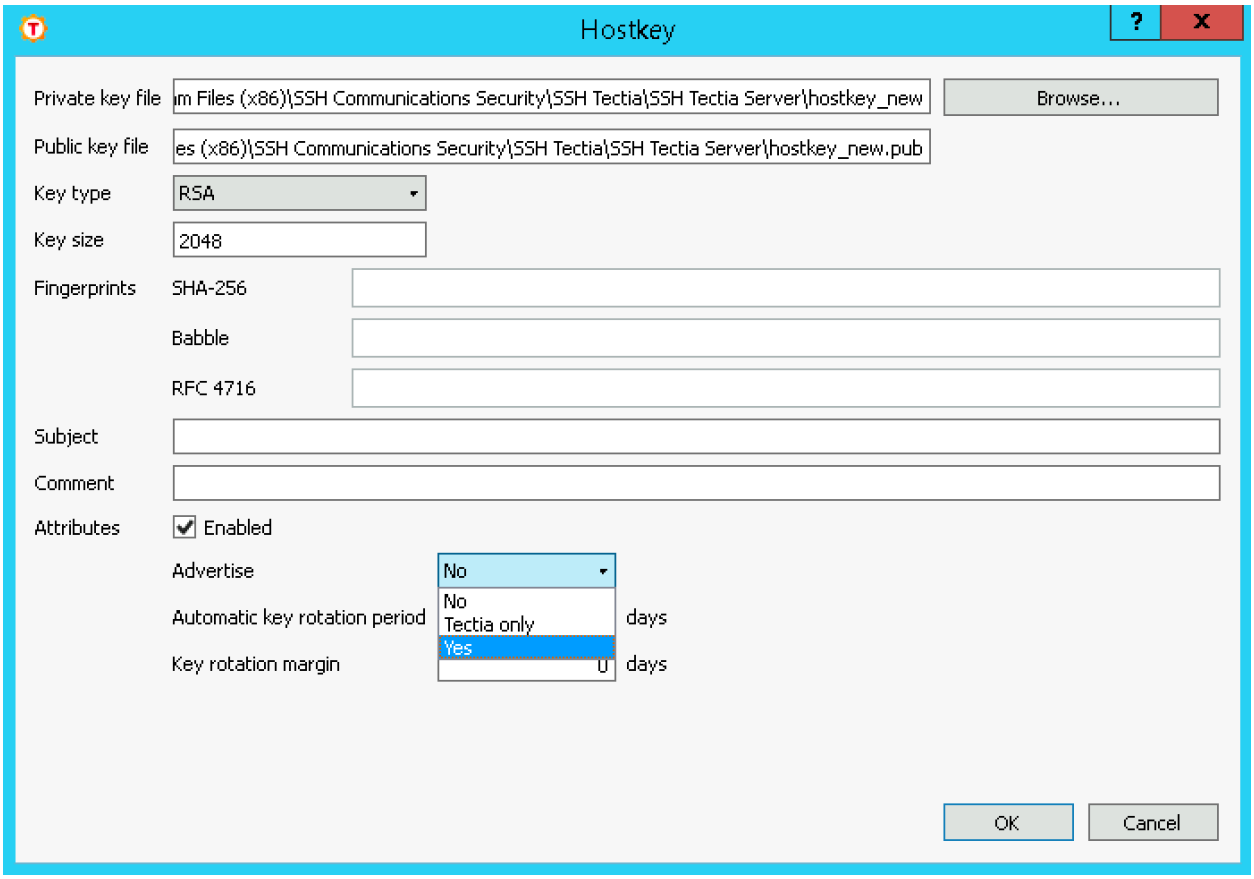
Tectia Server can be configured so that current RSA hostkey is enabled but not advertised and the new hostkey of different algorithm is both enabled and advertised. This allows secure shell clients that have connected before and support and enable Host Key Rotation / UpdateHostKeys to connect once and authenticate the server with RSA hostkey and after successful user authentication to add the advertised hostkey and remove the old RSA hostkey from known hosts within the same connection. For subsequent connections by this client the new hostkey is used provided that the client allows it in configuration.

If the same algorithm is used for the new host key, for example current hostkey is RSA and new RSA hostkey is generated, then only the first one is enabled as hostkey identity. In this case Tectia Server must be configured so that current RSA hostkey is enabled and advertised and the new RSA hostkey is advertised. This ensures secure shell clients that support and enable Host Key Rotation / UpdateHostKeys can connect and authenticate the server with current RSA hostkey and after successful user authentication add the advertised hostkey for future use when the old RSA hostkey is removed from Tectia Server.

Manual Rotation Example using RSA Host Keys

1. After upgrade create a new RSA host key “hostkey_new”

In Tectia Server Configuration GUI > Identity > Generate key



Or on command-line:

ssh-keygen-g3 -H -P hostkey_new

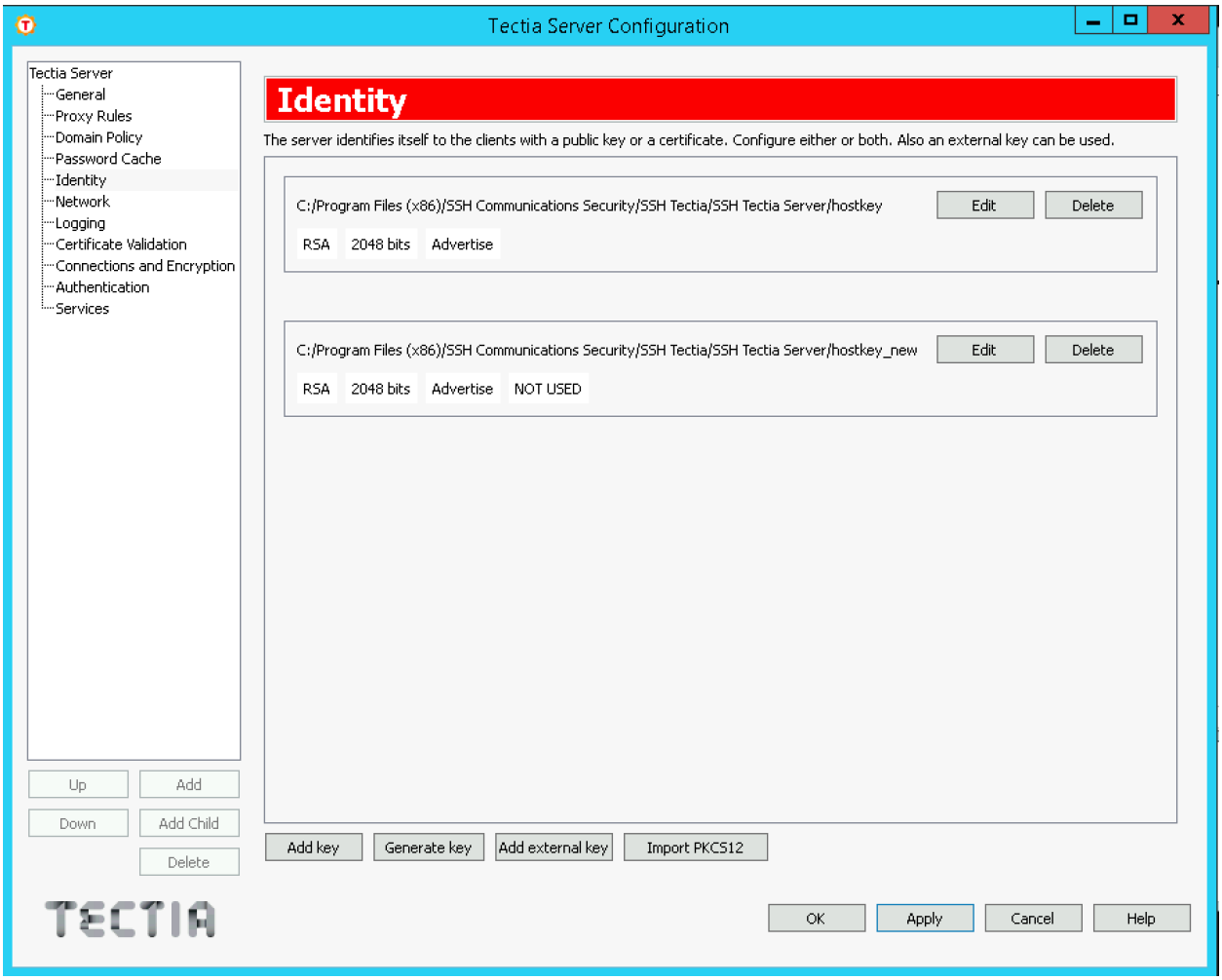
2. Advertise Current and New Host Key During Renewal Period Before Rotation

In Tectia Server Configuration GUI > Identity > Edit “hostkey” and “hostkey_new”

Change Advertise to “Yes” for both current host key and new host key so that they are advertised to secure shell clients and “Apply” to reload the configuration.

The current host key is used in server authentication and new host key is saved on the client-side for future use by the clients that support this.

Note that if current host key has Advertise set to “No” (default), then clients will remove it prematurely and the next connection will result in save host key prompt.



Or edit ssh-server-config.xml

```
 <hostkey advertise="yes">
  <private file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey" />
  <public file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey.pub" />
 </hostkey>
<hostkey advertise="yes">
  <private file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey_new" />
  <public file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey_new.pub" />
 </hostkey>
```

To reconfigure Tectia Server on command-line:

ssh-server-ctl reload

3. Monitor Tectia Server Logs

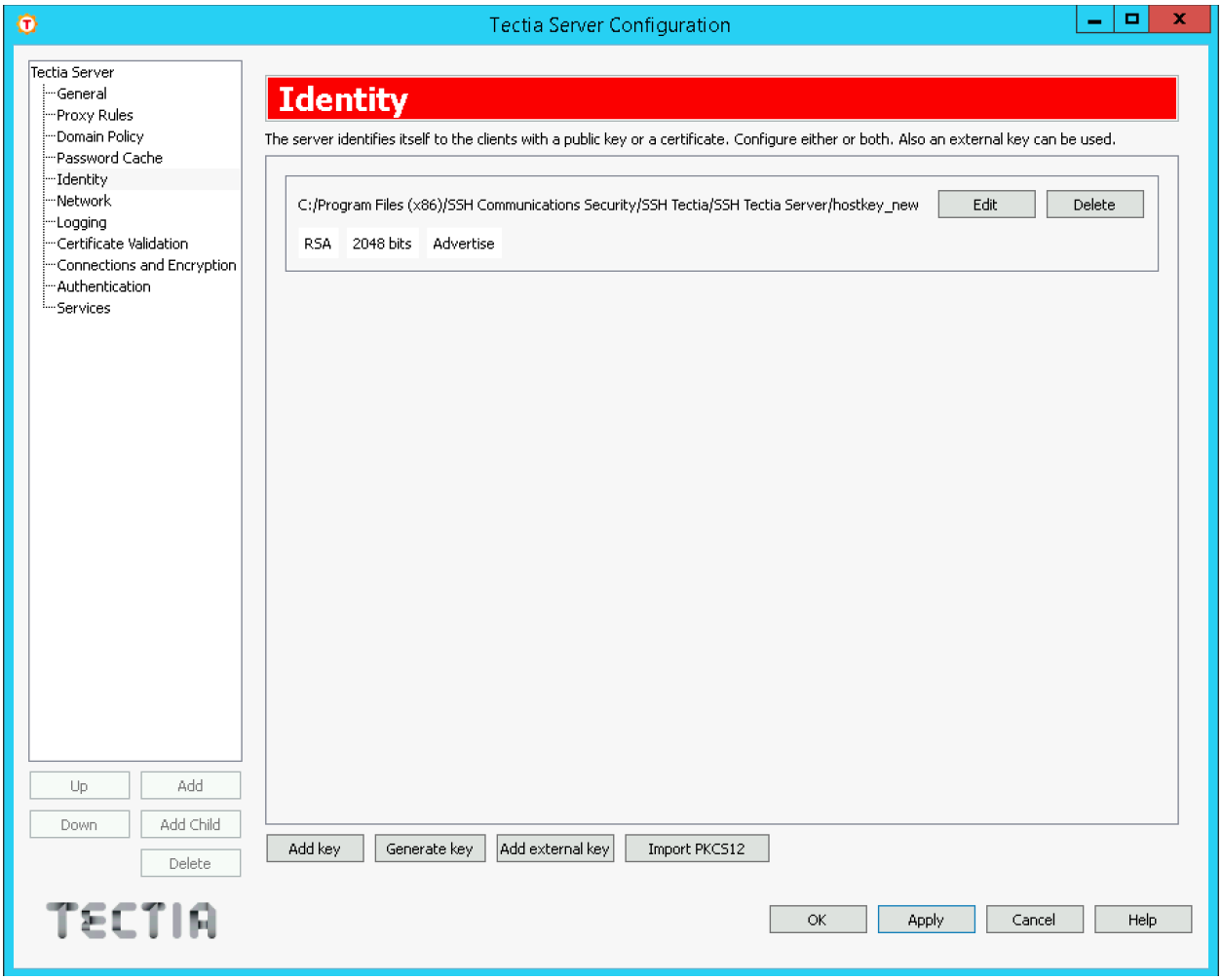
Tectia Server logs Hostkey-advert-accepted informational audit message when a client saves advertised new host key and server has proved ownership. These messages can be used to track the adoption of the new host key in the environment over time.

132 Hostkey_advert_accepted, Username: <authenticated_user>, Src IP: 127.0.0.1, Src Port: 50737, "xicaz-hepel-cemuf-kanid-kihyv-zesas-ralok-rinem-liluc-movuk-saxux", "SHA-256: gThQLwq2eiVIRd3T8X4JElNr9SGa7WRcUX93Audbe2I", Session-Id: 147, Protocol-session-Id: C85232F7554FBB5814ADAAD6D8E0C5FE7ECFE969F1AC754F75CCFB61FF044203

4. Host Key Rotation

When Tectia Server's old key is removed from configuration and new key taken into active use, any secure shell clients that have not connected during the renewal period or clients that do not support or do not have Host Key Rotation / UpdateHostKeys enabled and have previously connected and saved the old host key to known hosts fail to connect or prompt a host key changed warning.

In Tectia Server Configuration GUI > Identity > Delete "hostkey" to remove the old key from configuration and "Apply" to reload the configuration so that the "hostkey_new" becomes the current host key.



Or edit ssh-server-config.xml so that only “hostkey_new” key pair is specified:

```
<hostkey advertise="yes">
  <private file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey_new" />
  <public file="C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server\hostkey_new.pub" />
 </hostkey>
```

To reconfigure Tectia Server on command-line:

ssh-server-ctl reload

Rename “hostkey” to “hostkey_removed” and “hostkey.pub” to “hostkey_removed.pub” in installation directory so that it is not accidentally taken into use if Tectia Server is started without configuration file. Note also that next Tectia Server upgrade will automatically generate a new “hostkey” if the file with this name does not already exist.

If the current “hostkey_new” is advertised, then after successful user authentication clients that support and allow this in configuration will automatically remove from known hosts any host keys that are no longer advertised.

Tectia Client 6.4.19 has Host Key Policy Rotation enabled by default when connecting to Tectia servers only and Tectia Client attempts to remove the old keys from all known host key locations. On the client-side the following command can be used to view keys in local host key store(s):

ssh-keygen-g3 -F <host id> where <host id> is hostname or address#port

Fingerprints

The administrator can notify the users via some unalterable method of the expected fingerprint of the new host key and information when the new key will be taken into use.

The displayed fingerprint type on the client-side depends on the implementation and version of the client. For example recent Tectia Clients by default show both the Babble format and SHA256 base64 format fingerprints, recent OpenSSH clients show SHA256 base64 format fingerprint and PuTTY shows the RFC 4716 format fingerprint.

To obtain the fingerprints in Tectia Server Configuration GUI > Identity > Edit shows all three fingerprints of the current host key and any other host keys that are explicitly configured.

or

On the Tectia Server command-line

ssh-keygen-g3 --hash sha256 --fingerprint-type base64 -F hostkey_new.pub
ssh-keygen-g3 -F hostkey_new.pub
ssh-keygen-g3 --rfc4716 -F hostkey_new.pub
ssh-keygen-g3 --hash sha1 --fingerprint-type hex -F hostkey_new.pub

or

ssh-server-ctl status

Output shows SHA256 fingerprints for configured and any .next host keys used in automated rotation if enabled.

Replacing Host Public Key on Client-Side

For file transfer scripts or other non-interactive users, the public host key needs to be replaced on the client-side for clients that do not support hostkey rotation, for example file transfer jobs originating from Tectia SSH Server on IBM z/OS.

After the host key change client-side tools that obtain the current host key from the server like Tectia ssh-broker-ctl probe-key or ssh-keyfetch can be used. The following command can be used to view keys in local host key store(s) for the server:

ssh-keygen-g3 -F <host id> where <host id> is hostname or address#port, e.g. **serverhost**

z/OS Example

Verify the fingerprint automatically and replace the key, for example z/OS Tectia SSH Server version 6.6.9:

ssh-broker-ctl probe-key --hostkey-fp=<expected-fingerprint> --save-hostkey serverhost

The ssh-keyfetch tool can be used with Tectia SSH Server version 6.6.8 and below on IBM z/OS.

Windows Tectia Client Example

Replace the key hashed format and verify the fingerprint manually from output:

ssh-keyfetch --append=no -a -f hashed serverhost