1. Section 3.9, “Distributing SSH Keys for Remote Execution Manually” .
2. Section 3.10, “Using the Satellite API to Obtain SSH Keys for Remote Execution” .
3. Section 3.11, “Configuring a Kickstart Template to Distribute SSH Keys during Provisioning” .
4. For new Satellite hosts, you can deploy SSH keys to Satellite hosts during registration using the global registration template. For more information, see Registering a Host to Red Hat Satellite Using the Global Registration Template.

Satellite distributes SSH keys for the remote execution feature to the hosts provisioned from Satellite by default.

If the hosts are running on Amazon Web Services, enable password authentication. For more information, see https://aws.amazon.com/premiumsupport/knowledge-center/new-user-accounts-linux-instance.

3.9. DISTRIBUTING SSH KEYS FOR REMOTE EXECUTION MANUALLY

To distribute SSH keys manually, complete the following steps:

Procedure

1. Enter the following command on Capsule. Repeat for each target host you want to manage:

```
# ssh-copy-id -i ~foreman-proxy/.ssh/id_rsa_foreman_proxy.pub root@target.example.com
```

2. To confirm that the key was successfully copied to the target host, enter the following command on Capsule:

```
# ssh -i ~foreman-proxy/.ssh/id_rsa_foreman_proxy root@target.example.com
```

3.10. USING THE SATELLITE API TO OBTAIN SSH KEYS FOR REMOTE EXECUTION

To use the Satellite API to download the public key from Capsule, complete this procedure on each target host.

Procedure

1. On the target host, create the **~/.ssh** directory to store the SSH key:

```
# mkdir ~/.ssh
```

2. Download the SSH key from Capsule:

```
# curl https://capsule.example.com:9090/ssh/pubkey >> ~/.ssh/authorized_keys
```

3. Configure permissions for the **~/.ssh** directory:

```
# chmod 700 ~/.ssh
```

4. Configure permissions for the **authorized_keys** file:

```
# chmod 600 ~/.ssh/authorized_keys
```

3.11. CONFIGURING A KICKSTART TEMPLATE TO DISTRIBUTE SSH KEYS DURING PROVISIONING

You can add a **remote_execution_ssh_keys** snippet to your custom kickstart template to deploy SSH Keys to hosts during provisioning. Kickstart templates that Satellite ships include this snippet by default. Therefore, Satellite copies the SSH key for remote execution to the systems during provisioning.

Procedure

- To include the public key in newly-provisioned hosts, add the following snippet to the Kickstart template that you use:

```
<%= snippet 'remote_execution_ssh_keys' %>
```

3.12. CONFIGURING A KEYTAB FOR KERBEROS TICKET GRANTING TICKETS

Use this procedure to configure Satellite to use a keytab to obtain Kerberos ticket granting tickets. If you do not set up a keytab, you must manually retrieve tickets.

Procedure

1. Find the ID of the **foreman-proxy** user:

```
# id -u foreman-proxy
```

2. Modify the **umask** value so that new files have the permissions **600**:

```
# umask 077
```

3. Create the directory for the keytab:

```
# mkdir -p "/var/kerberos/krb5/user/USER_ID"
```

4. Create a keytab or copy an existing keytab to the directory:

```
# cp your_client.keytab /var/kerberos/krb5/user/USER_ID/client.keytab
```

5. Change the directory owner to the **foreman-proxy** user:

```
# chown -R foreman-proxy:foreman-proxy "/var/kerberos/krb5/user/USER_ID"
```

6. Ensure that the keytab file is read-only:

```
# chmod -wx "/var/kerberos/krb5/user/USER_ID/client.keytab"
```

7. Restore the SELinux context:

```
# restorecon -RvF /var/kerberos/krb5
```

3.13. CONFIGURING KERBEROS AUTHENTICATION FOR REMOTE EXECUTION

You can use Kerberos authentication to establish an SSH connection for remote execution on Satellite hosts.

Prerequisites

- Enroll Satellite Server on the Kerberos server
- Enroll the Satellite target host on the Kerberos server
- Configure and initialize a Kerberos user account for remote execution
- Ensure that the foreman-proxy user on Satellite has a valid Kerberos ticket granting ticket

Procedure

1. To install and enable Kerberos authentication for remote execution, enter the following command:

```
# satellite-installer --scenario satellite \
 --foreman-proxy-plugin-remote-execution-ssh-ssh-kerberos-auth true
```

2. To edit the default user for remote execution, in the Satellite web UI, navigate to **Administer** > **Settings** and click the **RemoteExecution** tab. In the **SSH User** row, edit the second column and add the user name for the Kerberos account.

3. Navigate to **remote_execution_effective_user** and edit the second column to add the user name for the Kerberos account.

To confirm that Kerberos authentication is ready to use, run a remote job on the host.

3.14. SETTING UP JOB TEMPLATES

Satellite provides default job templates that you can use for executing jobs. To view the list of job templates, navigate to **Hosts** > **Job templates**. If you want to use a template without making changes, proceed to Executing a Remote Job.

You can use default templates as a base for developing your own. Default job templates are locked for editing. Clone the template and edit the clone.

Procedure

1. To clone a template, in the **Actions** column, select **Clone**.

2. Enter a unique name for the clone and click **Submit** to save the changes.

Job templates use the Embedded Ruby (ERB) syntax. For more information about writing templates, see the Template Writing Reference in the *Managing Hosts* guide.