

---

# ReST Installer User's Guide

An introduction to the ReST Installer.

Jeff M Larkin, Innovative Computing Laboratory, UT  
<larkin@cs.utk.edu>

Eric T Meek, Innovative Computing Laboratory, UT  
<meek@cs.utk.edu>

Revision History

3/11/2005

Initial Document Release.

Revision 1.0

## Abstract

This document is written as a basic introduction to using the ReST Installer. No prior knowledge of the ReST Installer is needed.

## Table of Contents

ReST Installer Introduction .....	1
Basic Usage .....	2
Setup Locations .....	2
Setup Logical Groups .....	2
Setup Actions .....	4
Setup Authentication .....	4
The Installation .....	6
Advanced Topics .....	6
Setup File Substitutions .....	6
Configuring Command Options .....	6
Saving an Install Script .....	6
Examples .....	7

## ReST Installer Introduction

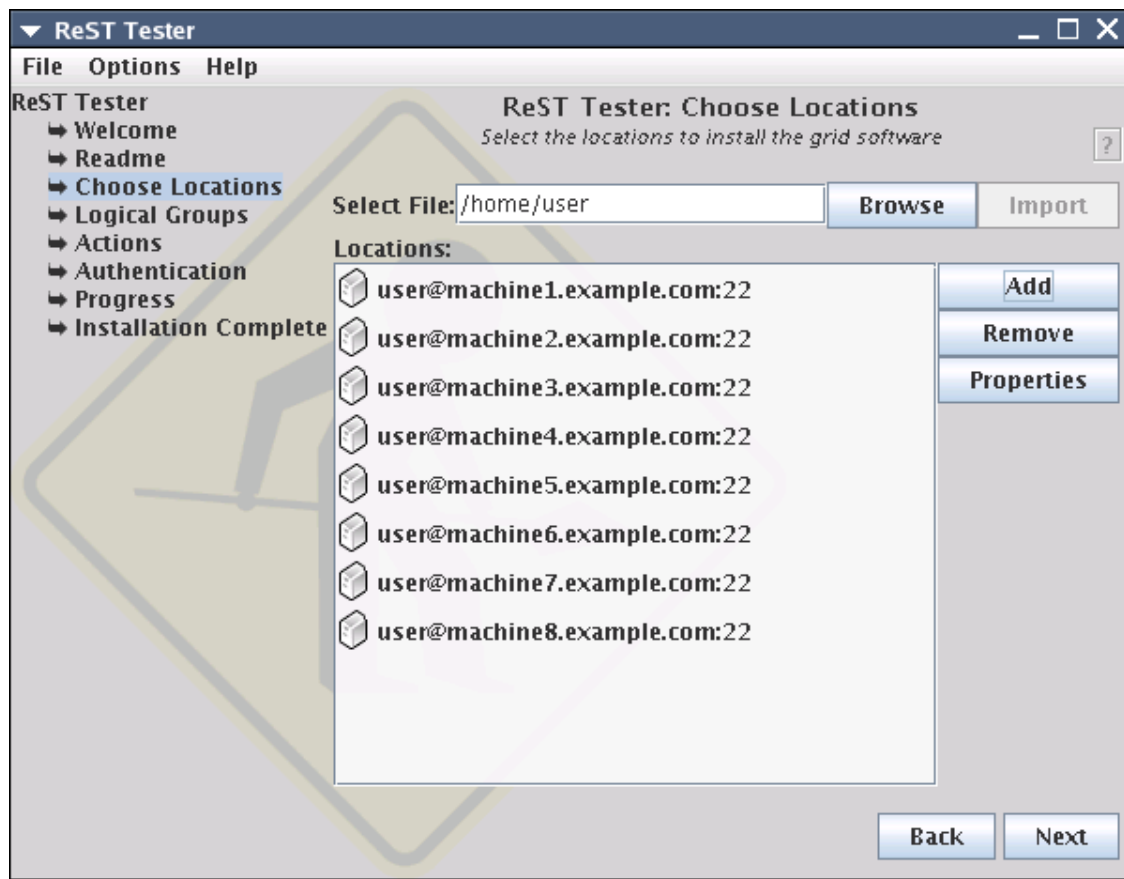
For consumer software installation, Installation "Wizards" have become the de-facto standard; but in a distributed computing environment, such a luxury is almost nonexistent. Software installation in a distributed environment often require a user to login to numerous machines and perform roughly the same tasks on each. Further compounding the problem, most source distributed software installation requires both a lengthy and confusing configuration, compilation, and installation process. The ReST Installer provides a familiar, wizard-like interface to source and binary distributed software that can be used to install software in a heterogeneous environment easily and automatically. At the core of the ReST Installer is the ReST package model. By defining the structure of the ReSI software packages, the ReST package model provides software maintainers a way to define their software's requirements and options so that they can be presented to a user in a simplified manner, shifting the burden of software installation from the user to the software provider. Generic instructions on installing software with the ReST Installer are contained in the sections below. Developers interested in using ReST to distribute their software should instead read the *ReST Package Maker's Guide*.

# Basic Usage

## Setup Locations

The Setup Locations screen allows for choosing a one or more locations on which to install the package. The user must have a valid login for each machine listed and the machine must be reachable via secure shell (SSH). Locations can be inserted using a regular expressions to define multiple locations with similar names. For example, entering a location as `someone@somewhere[0-2].example.com` would create three locations of `someone@somewhere0.example.com`, `someone@somewhere1.example.com`, `someone@somewhere2.example.com`. If a group of machines will be used for the installation of several packages, the user may wish to write a script file containing the list of machines. An example of such a file exists later in this document.

**Figure 1. Setup Locations**



This screenshot shows the setup locations dialog of the ReST Installer. This user has setup 8 locations on which the software should be installed.

## Setup Logical Groups

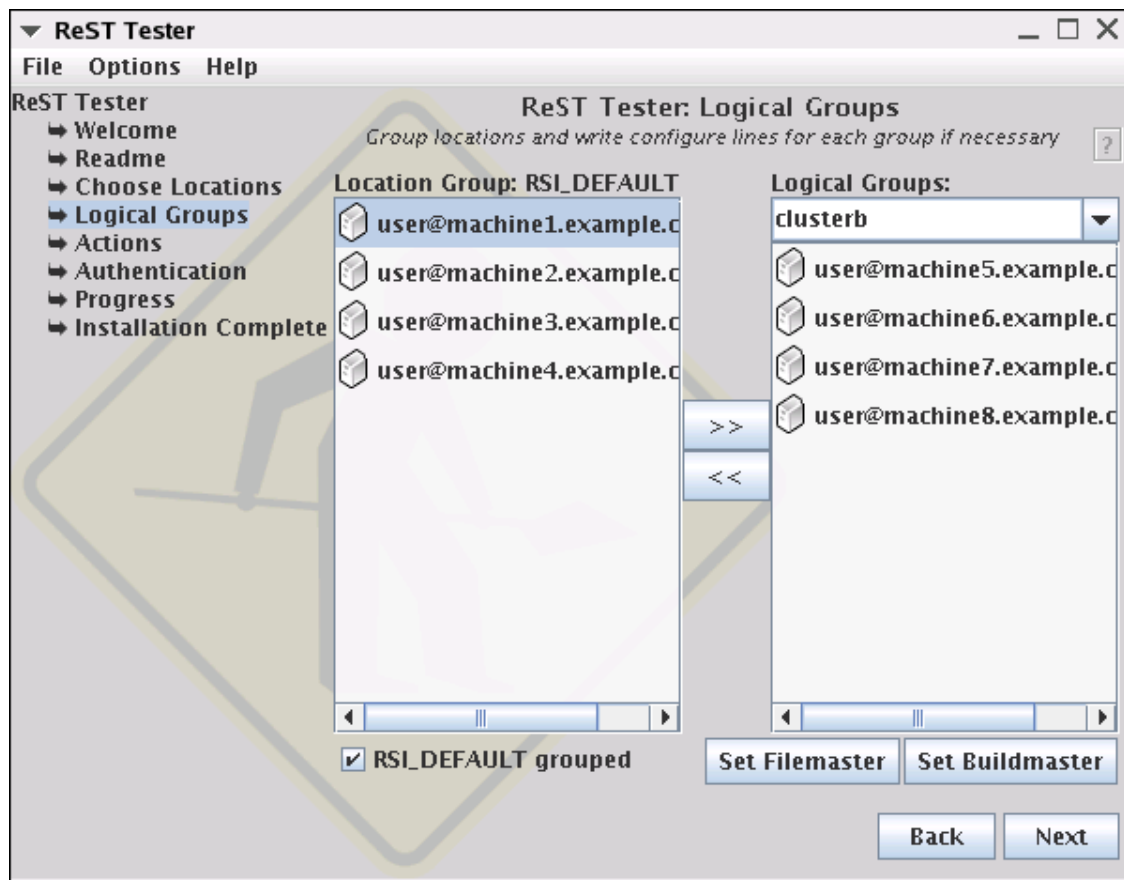
A logical group should consist of locations grouped by filesystems and binary compatibility. In other words, if five machines share a common filesystem and programs compiled on one machine may be run on all, then these machines should be placed in a logical group. By creating logical groups, the Rest In-

staller is able to reduce the network traffic, file space and time needed to install the package. Two important entities exist among logical groups: build masters and file masters. When installing a package, some parts of the package may only need to be executed on one machine for every machine in the group to benefit from that action. In this case the command will only be run on the build master of the group. Every group must have a build master. If no build master is explicitly selected by the user, one will be automatically chosen from the group. Once a build master is selected, a single asterisk will appear by its name. To explicitly choose a build master, select one machine in the group and press the *Set Buildmaster* button below the group list.

Multiple logical groups may share a common filesystem, making it possible to transfer files to one machine and have them available to machines within multiple logical groups. For this reason file master entities exist. A file master is an extension of a build master, so if a machine is a file master, it is also the build master of its logical group. To select a filemaster, choose a machine and press the *Set Filemaster* button below the groups list. This will cause a window to appear, which will include all of the defined groups. To select multiple groups within this window, hold the *ctrl* key while clicking on the group name. When you have selected the desired groups, press *Ok*. Selecting a file master will only affect the build master of the group to which the new file master belongs. All other groups will remain unchanged. When a machine has been selected as a file master, two asterisks will appear by its name.

By default a logical group is created for all machines unselected in the groups panel. This group is known as `RSI_DEFAULT` and may be treated as a normal logical group or treated as separate machines, each in their own logical group. By default the `RSI_DEFAULT` group is treated as a logical group. To treat each machine as an independent group, simply uncheck the *RSI\_DEFAULT grouped* checkbox on the groups panel.

**Figure 2. Setup Logical Groups**

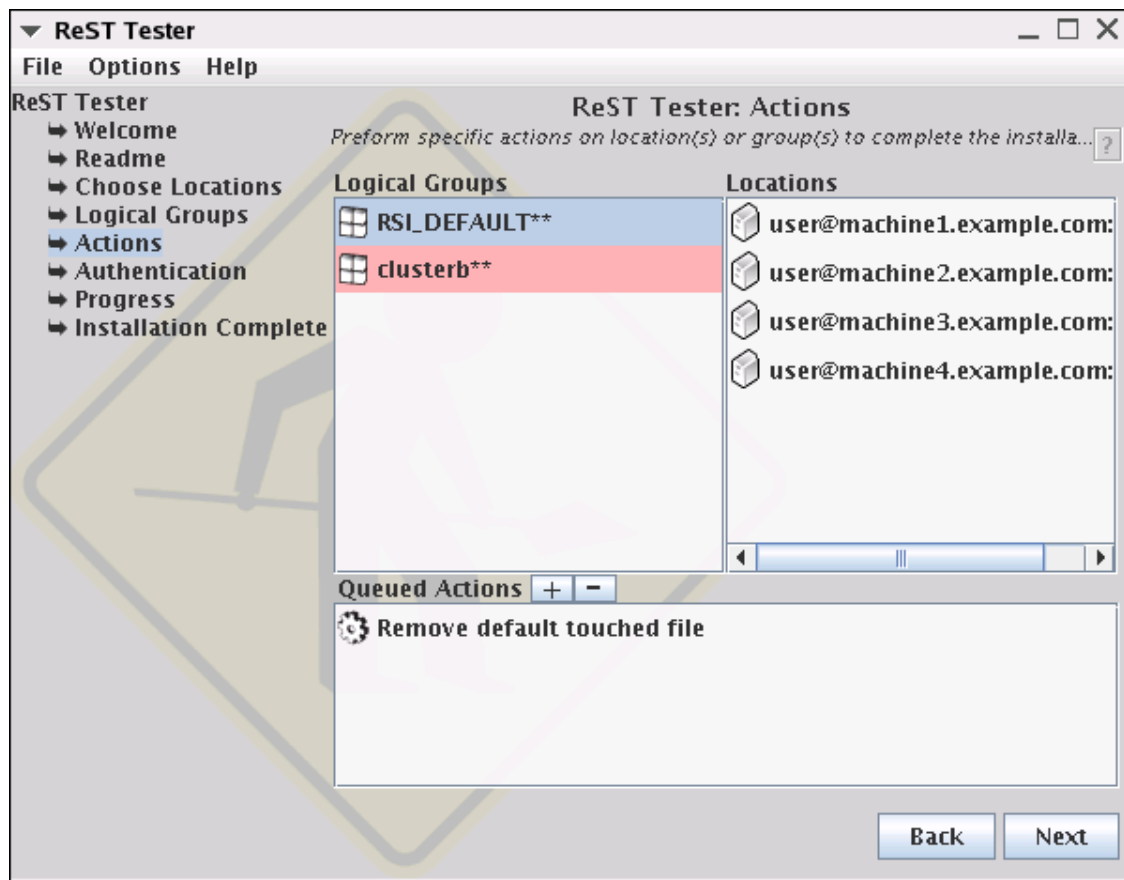


This screenshot shows the setup logical groups dialog of the ReST Installer. This user has split the 8 machines into 2 groups by selecting a group for 4 machines, while leaving the remaining machines in the RSI\_DEFAULT.

## Setup Actions

Actions can be added to either a location or a logical group. Adding an action to a logical group will cause the action to be performed on each location in the group and is very useful in adding actions to many locations quickly. To add an action to a location or logical group, select the target of the action, click on the + button and then select the action(s). To remove actions, select the group or location with the actions, select the actions and then click on the - button.

Figure 3. Setup Actions



This screenshot shows the setup actions dialog of the ReST Installer. This user has added the *Remove default touched file* action to the RSI\_DEFAULT group.

## Setup Authentication

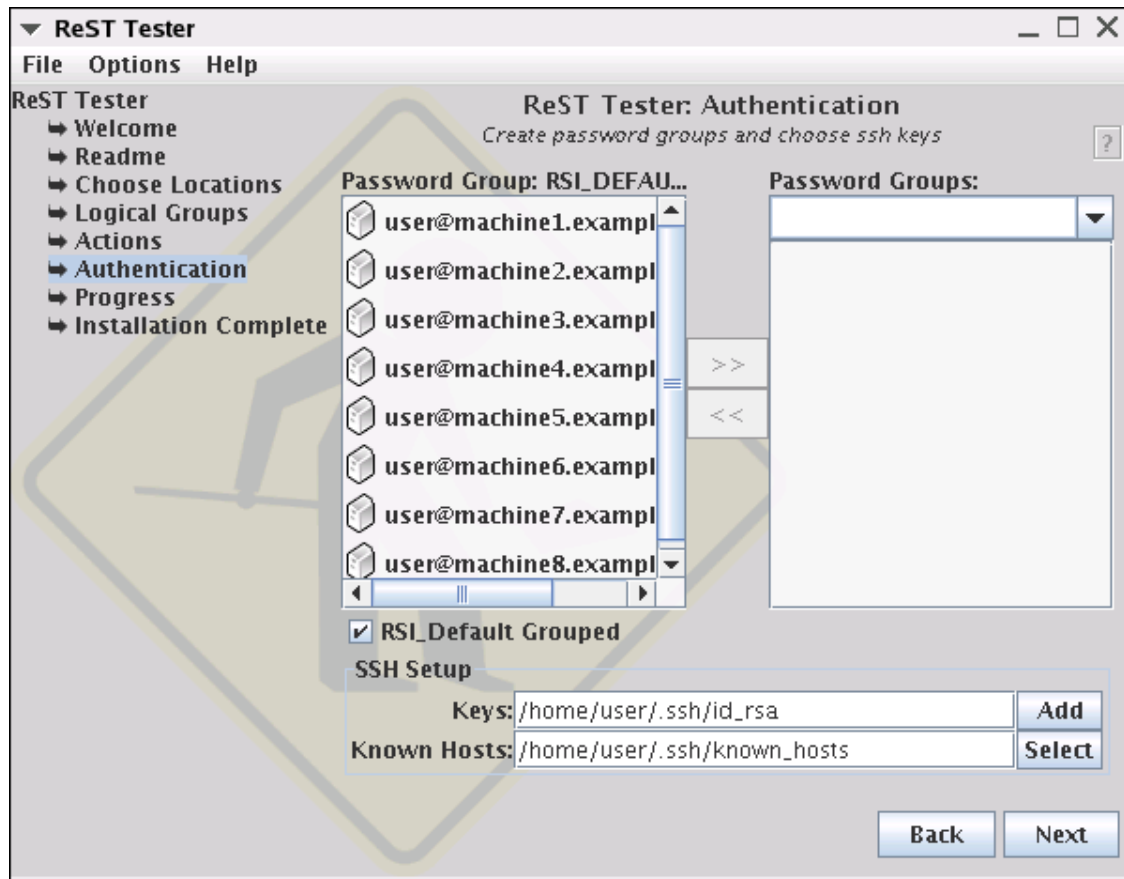
The final step before software installation is configuring how the ReST Installer will access the remote machines. At this time ReST can only access machines via Secure Shell (SSH). The preferred authentication method is via public key authentication. If you have SSH keys that can be used to authenticate with the remote machines, press the *Add* button next to the keys field. This will allow you to select one

or more OpenSSH formatted private key files to use for authentication. If you have used the ReST Installer with SSH keys before, the keys you have previously used will already appear in this field. It is important to note that any key listed in the text field will not be removed when adding a new key. If a key is to be replaced, first it's path must be manually removed and then the new key must be added using the add button.

If key-based authentication is not an option, users may create password groups from the listed machines. A password group contains machines that require the same password for remote access. Creating password groups is done in the same way as creating logical groups; simply select the desired machines and press the >> button to move the machines into the new group, then just name the group for convenience. Now when the Installer tries to access the machines in this group, you will only be asked for the password once. Be warned, however, that this convenience is achieved by caching the password to memory, which is a potential security hazard. Users who do not want this behavior should use SSH keys for authentication.

The final step of configuring authentication is to point the ReST Installer to your current known\_hosts file. When SSH makes a connection to an unknown machine it caches the server's key in a known hosts file to compare against at the next login. If you already have a known hosts file, use the *Select* button to choose your known hosts file. If you do not have one, simply enter the name of the file that you would like to use. By using a known hosts file that has already been populated with keys, you avoid being asked to authorize the keys of the remote machines.

**Figure 4. Setup Authentication**



This screenshot shows the setup authentication dialog of the ReST Installer. This user has provided the location of an SSH key and known\_hosts file.

## The Installation

Once you have given the needed authentication credentials, you will see the list of locations and the status of their installation. The icon to the left of the location name will show the machine status, either non started, in progress, warning, error, or completed. To the right of the location name appears what is currently being performed on the location or an error message. Once all the locations have been attempted and have either completed or failed, the next button will become available. Press the next button to complete the installation.

If an error occurs during the installation process it may be useful to view the output generated on the location. ReST saves the output of the previous installation to an XML file so that the user may attempt to learn why failures occur. The output file is saved in the user's home area in the subdirectory *.rest/output/LOCATION.xml*, where *LOCATION* is the shorted hostname of the desired location. Windows users should know that the *.rest* directory will appear in *C:\Documents and Settings\USERNAME*. A web-based utility has been provided to give users an easy way to view this output file. This utility can be found at <http://icl.cs.utk.edu/rest/> under the *Tools* menu item.

If the installation completed without an error, the software can be found in the *rest/* subdirectory of the user's home area. At this time there is no way to change the installation directory for rest packages, although this feature will be added in later versions. Currently the software is installed in *HOME/rest/GROUPNAME/PACKAGEBASE/*. For example, the BLAS package has a package base of *libs*, so if the install was configured for a group named *group1*, then the library would be found in *HOME/rest/group1/libs*.

## Advanced Topics

By default the ReST Installer runs in simplified mode, which removes the need for users to configure specific package options. In this mode packages are installed using defaults created by the software packager. Often times, however, it is desirable to configure the software for the machine on which it is being installed. To allow advanced configuration of packages selected *Advanced* from the *Options* menu. You should notice two new steps to in the tree along the left of the installer: File Substitutions and Configure Commands. An explanation of each of these steps comes below.

### Setup File Substitutions

Some packages require certain configuration files to be edited before the package will run correctly or with special options enabled. The ReST Installer will display these files as a form that can be quickly edited. Options that are either on or off are represented by a check box. The option may be a list of possible choices, which are represented by a drop-down box. If multiple choices can be selected, this is achieved by holding the *ctrl* key while selecting choices. Some options may require text input and are represented by either a single-line text box or a multiple-line text area. The process of editing the configuration files should be familiar to anyone who has filled-out a form on a web page. At this time file editing is done on a per-group basis and the current group can be selected from a drop-down list next to the *reset* button.

### Configuring Command Options

Configuration of command options is done almost exactly like editing configuration files. It is worth noting that options requiring text input are represented by text boxes and that an empty text box signifies that the option is disabled. To enable a text option simply click in the text box and add text. To disable a text option, simply delete the content of the text box. Just as with configuration files, command options are group-based and the current group can be selected from the drop-down list beside the *reset* button.

### Saving an Install Script

The ReST Installer allows users to create installation scripts to duplicate the last installation at some time in the future. This is useful for packages that update frequently, allowing the user to step through the installation without configuring groups or options. After installation is complete, the ReST Installer will automatically ask you if you'd like to save an installation script upon closing the Installer. Once you have saved a script, you can use it in future Installer sessions by selecting it on the *Setup Locations* screen of the Installer and choosing *Import*.

It should be noted, for advanced users it is possible, for several groups to be configured using similar options and/or substitutions by configuring the first group as desired and then save the install. Before the install is saved, all the previous panels should be completed as desired so the saved install is faithfully recreated. After saving the install, the generated script can be manually edited making duplication of group options and substitutions as easy as a cut and a paste.

## Examples

### Example 1. Remote Machine List File

```
groups:
group1
group2

locations:
user@machine[1-4].example.com:22 group1
user@machine[5-8].example.com:22 group2
filemasters:
user@machine1.example.com:22 group1 group2

buildmasters:
user@machine1.example.com:22 group1
user@machine5.example.com:22 group2

actions:
```

This is a simple example of a script that when loaded will create two groups, each containing four machines. These groups have a common file master. It is possible to create more complex scripts to completely recreate installer runs, but this example simply imports a list of machines to use in the install.