http://www.vmware.com/files/templates/images/bg-wrapper.gif



*Prepared for:*



*Prepared by:*

David Gress

gressd@vmware.com

**.2**

EPV-Linux-Files PackageType

**PREPARED BY:**

|  |  |  |
| --- | --- | --- |
| **Name and Project Role** | **Company/Title/Department** | **Date** |
| David Gress | VMware /Architect/ Custom Engineering | 19 April 2017 |

**ENGINEERING REVIEW:**

*Signature(s) here indicate that the individuals have read and verified the content of this document.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name and Project Role** | **Company/Title/Department** | **Signature** | **Date** |
|  |  |  |  |
|  |  |  |  |

**CUSTOMER APPROVED BY:**

*Signature(s) here indicate that the individuals have read and verified the content of this document has been delivered through delivery and custom development and integration.*

*Signature indicates the acceptance of deliverables associated with Phase 1.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name and Project Role** | **Company/Title/Department** | **Signature** | **Date** |
|  |  |  |  |
|  |  |  |  |

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name and Project Role** | **Company/Title/Department** | **Comment** | **Date** |
| David Gress | VMware/ PSO | Initial Design & Development | 19 April 2017 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[Revision History 3](#_Toc480481818)

[Purpose and Scope 6](#_Toc480481819)

[Glossary of Terms 6](#_Toc480481820)

[Reference Documents 6](#_Toc480481821)

[Assumptions and Open Issues 6](#_Toc480481822)

[Technical Assumptions 6](#_Toc480481823)

[Business Assumptions 6](#_Toc480481824)

[Open Issues 7](#_Toc480481825)

[Introduction 7](#_Toc480481826)

[Install package: com.vmware.pso.epv.linux.files 7](#_Toc480481827)

[vRCS adding EPV Linux endPoints 8](#_Toc480481828)

[Add Content Endpoint 8](#_Toc480481829)

[Endpoint Policy Settings 9](#_Toc480481830)

[Connection Test 10](#_Toc480481831)

[Request Data 11](#_Toc480481832)

[EPV-Linux-Files Configuration 12](#_Toc480481833)

[Capture 12](#_Toc480481834)

[basePaths - endpoint Config 12](#_Toc480481835)

[Capture EPV-Linux-Files Example 14](#_Toc480481836)

[Request a Capture 14](#_Toc480481837)

[Monitor the request 17](#_Toc480481838)

[Day 2 Operations 18](#_Toc480481839)

[Test Package 18](#_Toc480481840)

[Release Package 19](#_Toc480481841)

[Schema – Flow logic 21](#_Toc480481842)

[List 21](#_Toc480481843)

[Capture 21](#_Toc480481844)

[Release to endPoint 22](#_Toc480481845)

[Capture & Test 23](#_Toc480481846)

[Initiate Pipeline 23](#_Toc480481847)

[Monitor Results 26](#_Toc480481848)

# Purpose and Scope

This document contains the requirements, usage and development overview of the implementation of capture and testing of Linux Files on endpoint hosts where the password is controlled with EPV. This package type is a not supplied with base codestream/Houdini and is a custom type developed to meet the requirements for capture/test/release where EPV controls the passwords on the servers (endpoints).

# Glossary of Terms

|  |  |
| --- | --- |
| **Term** | **Explanation** |
| vRCS | vRealize CodeStream |
| RCS | Remote Content Server |
| vCO | vRealize Orchestrator |
| vRO | vRealize Orchestrator |
| EPV | Enterprise Password Vault |
| Houdini | vRealize Code Stream Management Pack for IT DevOps |
| endPoint | The remote system being configured for the task |

# Reference Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Document** | **Revision** | **Date** |
|  |  |  |  |

# Assumptions and Open Issues

## Technical Assumptions

* The capture/release of Linux files will be driven by vRCS executing the newly created PackageTypes of EPV-Linux-Files.
* Configuration setup will be done to match requirements when the package is installed.

## Business Assumptions

* Current working vRCS setup within the environment.
* EPV installed, configured and tested
* If work has begun on a requirement and Pfizer decides to change the requirement while work effort has begun, VMware will invoice for the work performed to date
* This document is not intended to bond the requirements to a time table, schedule or specific due date VMware will not provide additional functionality or functionality considered an assumption to a given requirement. The functionality defined in this document will be the functionality used to design and implement the requirement

## Open Issues

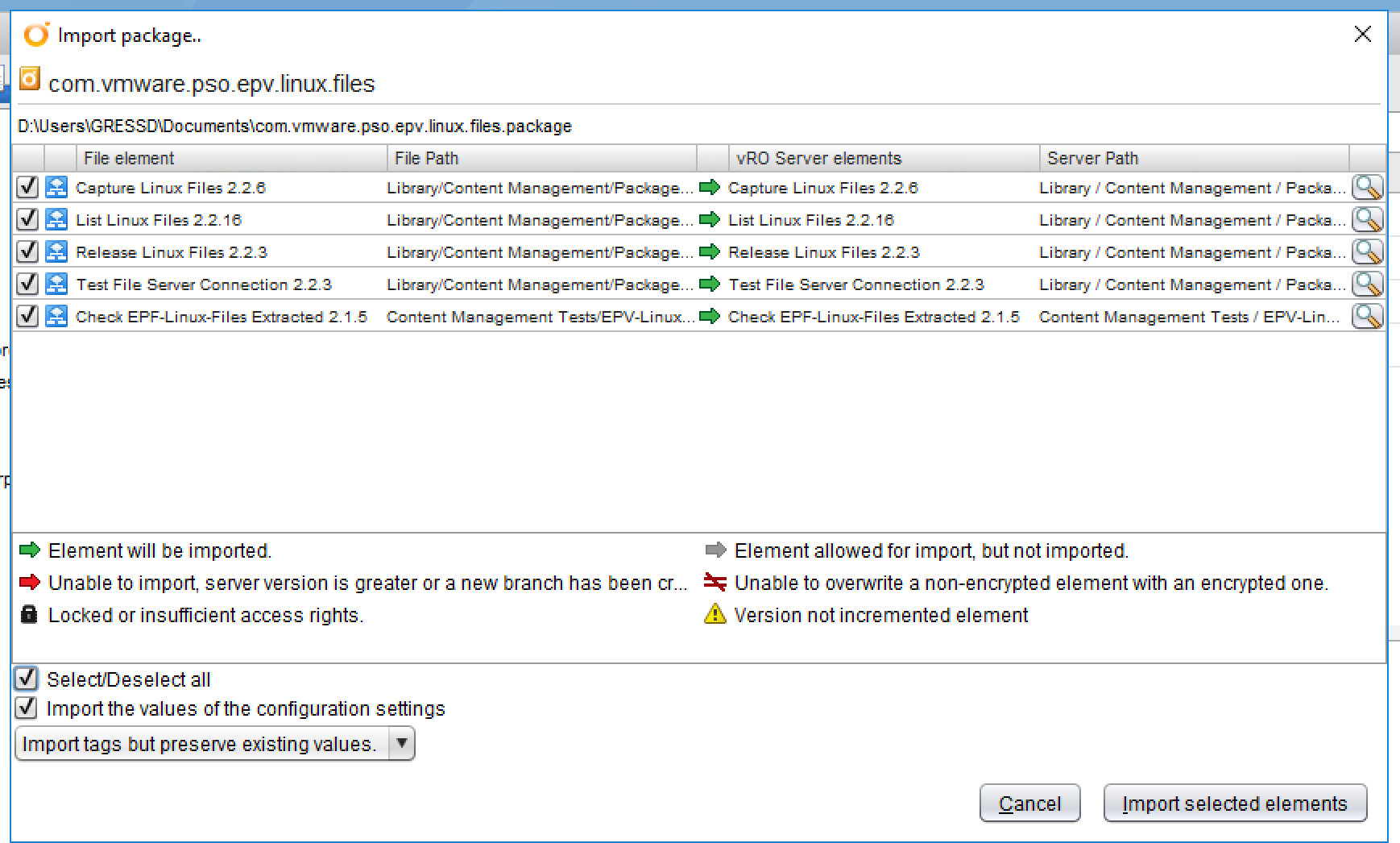
* Each Linux endpoint (server) for Capture/Test/Release will be registered within the CyberArk EPV system.

# Introduction

Currently the collection and distribution of Linux for release management is a manual process to copy and distribute the said files (artifacts). The automation of this process via vRCS will also capture the user/group and file or directory mod attributes to allow compatibility across systems.

# Install package: com.vmware.pso.epv.linux.files

1. Using vRO in Design mode, import the supplied EPV-Linux-Files package into the vRCS system.
2. Ensure that all files are selected



vRO Install Location

When the package is installed, the workflows are installed in the following location:

Library / Content Management / Package Types / Other /

EPV-Linux-Files

Library / Content Management Tests / Package Types / Other /

EPV-Linux-Files / Other /

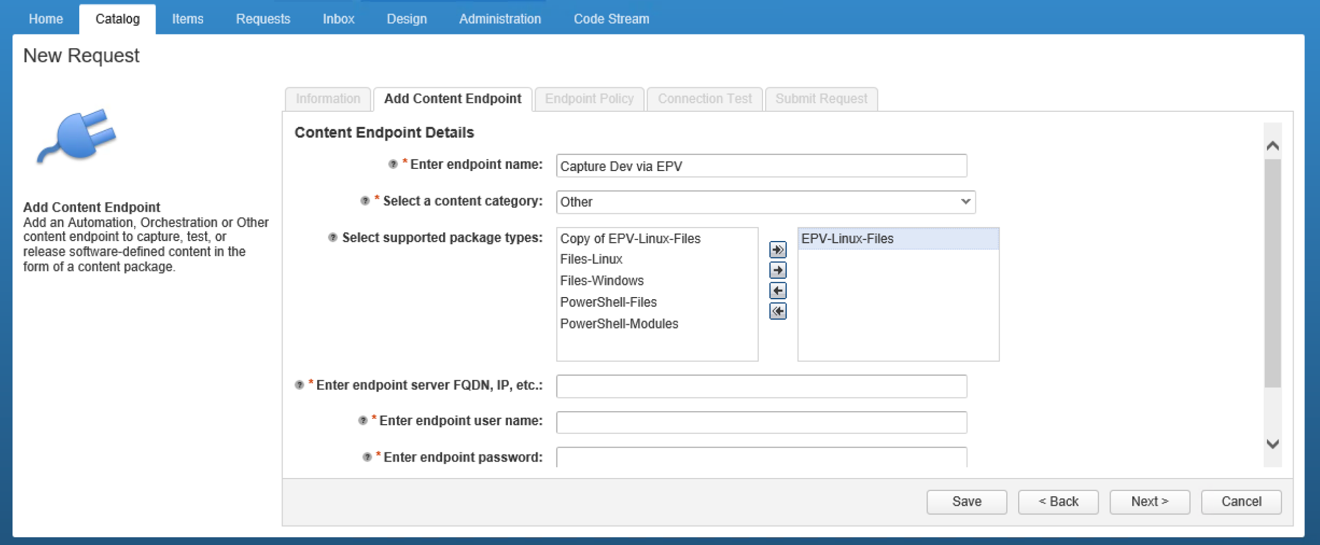
# vRCS adding EPV Linux endPoints

In order to use vRCS to capture/test/release linux artifacts between systems, you must first configure the endpoints with the appropriate user name as the password is obtained when the pipeline runs. The “Type” of action allowed for that endpoint must also be configured at this stage.

### Add Content Endpoint

* This process is the same for either PowerShell-Files or PowerShell-Modules. The selection below specifies which type will be support for the endpoint.

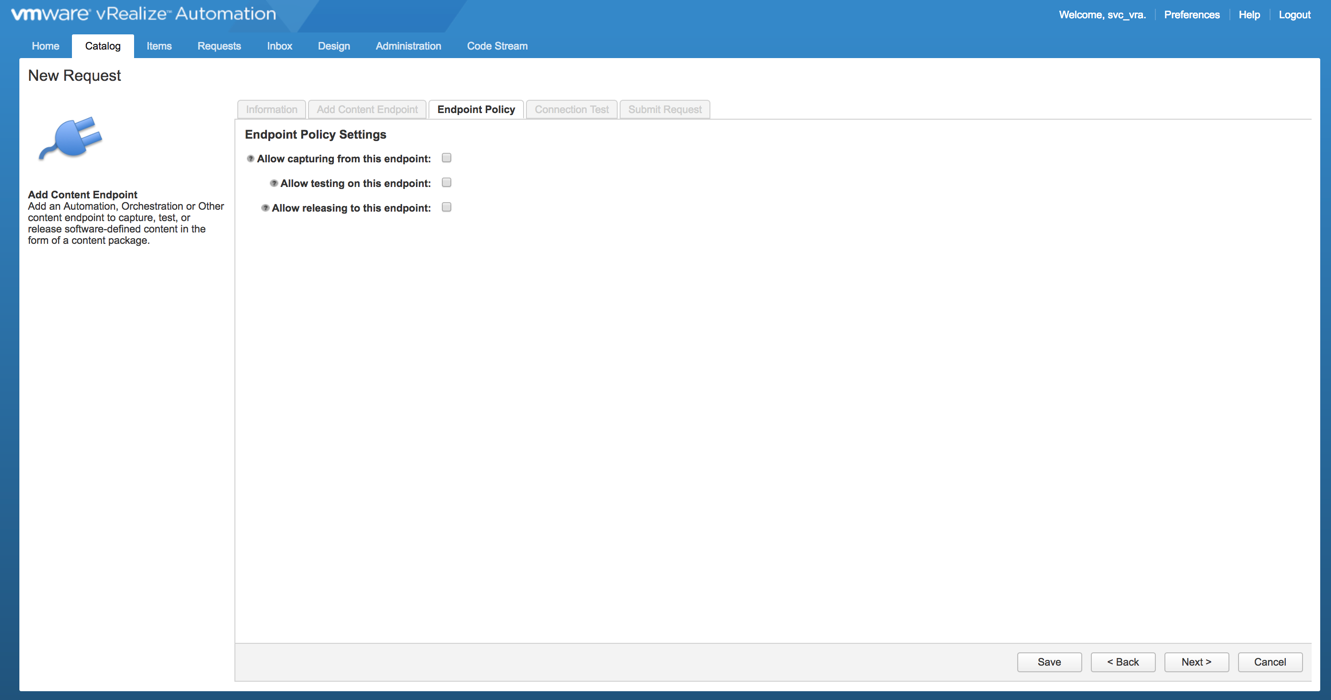
1. Login to vRCS
2. Select the Catalog Item: Add Content Endpoint and hit Next
3. Fill out the form:
   * 1. endpoint name: Give a unique name that describes the endpoint
     2. Select category: Select Other
     3. For each type of artifact to support EPV-Linux-Files, highlight and move to the right as seen below
     4. Endpoint FQDN: Enter the endpoint FQDN, this must match what EPV has stored in it’s database.
     5. Endpoint UserName: Enter the username that has Read access for Capture and ReadWrite access for Test or Release
     6. Endpoint Password: Enter anything here, it is not used.
     7. Endpoint Tag: Optional, group multiple endpoints together with tags



* Hit Next

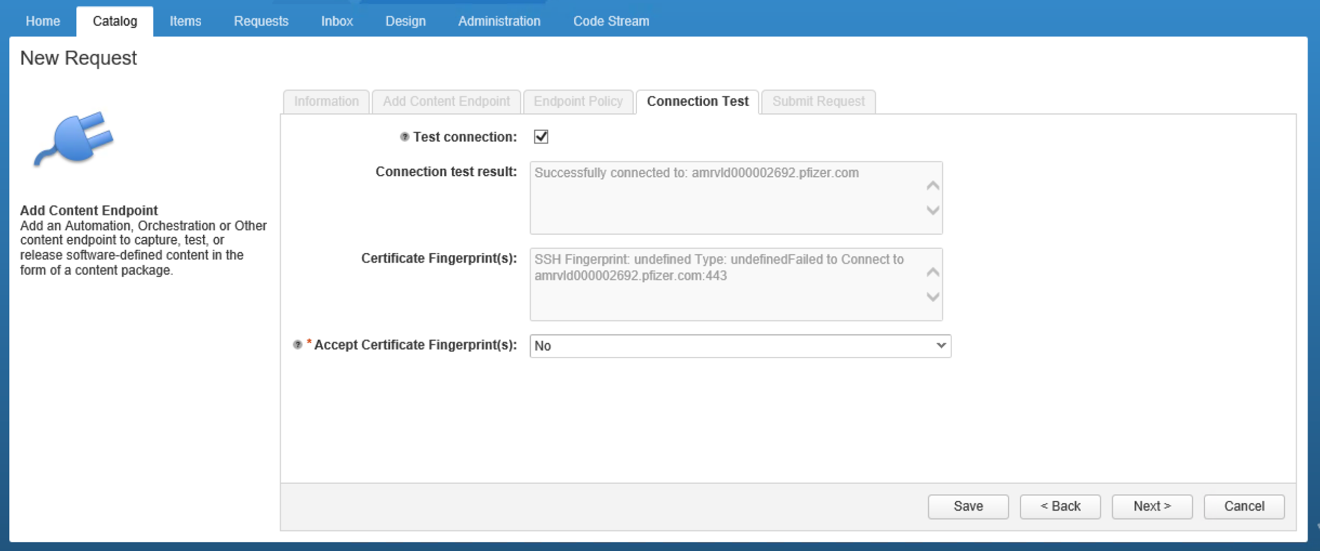
Endpoint Policy Settings

* 1. Allow Capture: Select this if artifacts will be captured from this endpoint
  2. Allow Testing: Select this if this endpoint supports testing of the artifacts
  3. Allow Release: Select this if this endpoint allows releasing of artifact to it, this is the production endpoint.
  4. Hit next



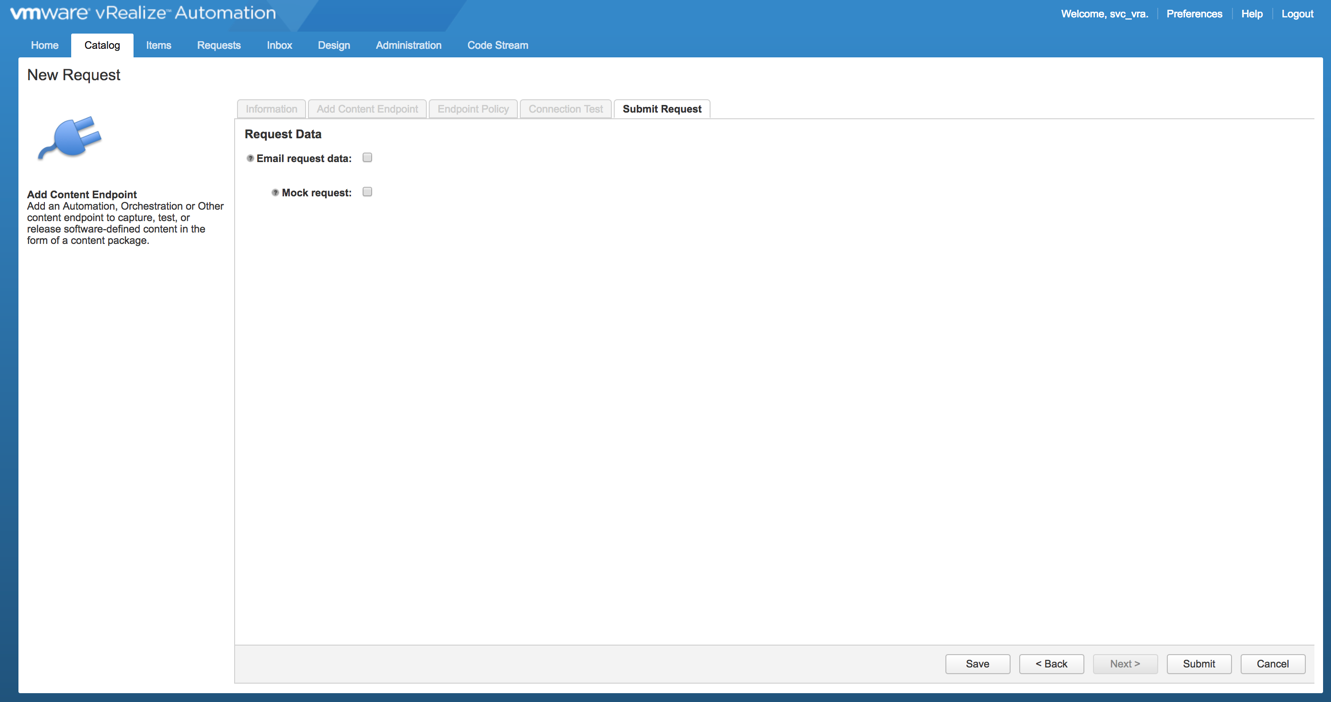
Connection Test : Verifies credentials supplied and certificate acceptance

* 1. Click the “Testi Connection “ box
     1. The results should be “Successfully connected to <endpoint>”
  2. Accept Certificate Fingerprint : Drop down and select “No”
     1. It is OK to see the “SSH Fingerprint : Undefined” here
  3. Hit Next



### Request Data

1. If you would like to receive a copy of the request via email click the “Email Request Data” and enter your email address
2. The mock request is used to test items, the endpoint will NOT be added if this is checked, this can be used for testing prior to addition.
3. Hit Submit
   1. The request will be queued and processed by vRA, when complete the endpoint is configured for use



# EPV-Linux-Files Configuration

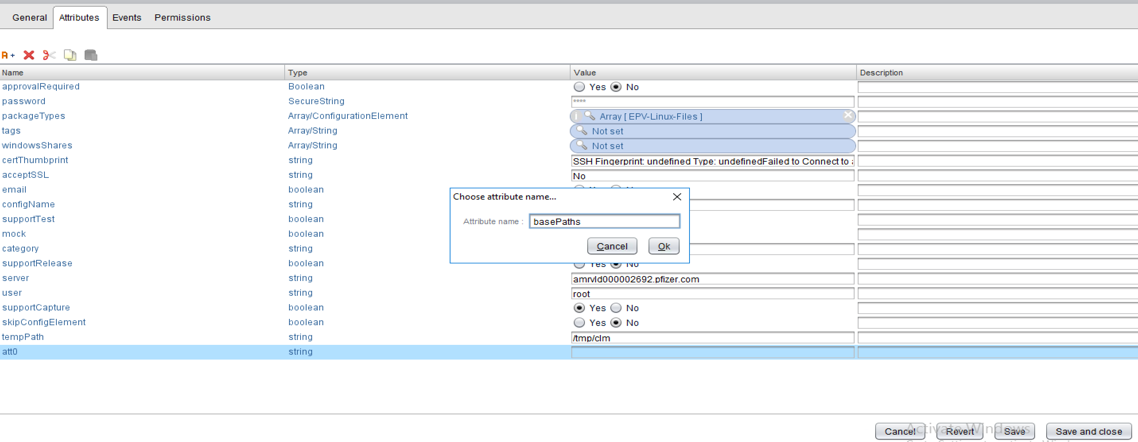
After the endpoint is configured, setup the configuration for processing. The configuration element basePaths will need to be modified for each file or directory which is requested to be captured.

## Capture

### basePaths - endpoint Config

The endpoint which was added that artifacts are allowed to be will need to be modified manually for the setup in the environment for which files or directories are available for the Capture process. Below is a sample of the configuration when the endpoint is added. Edit the configuration and add a “basePaths” string attribute.

1. Create the attribute named basePaths as a string variable

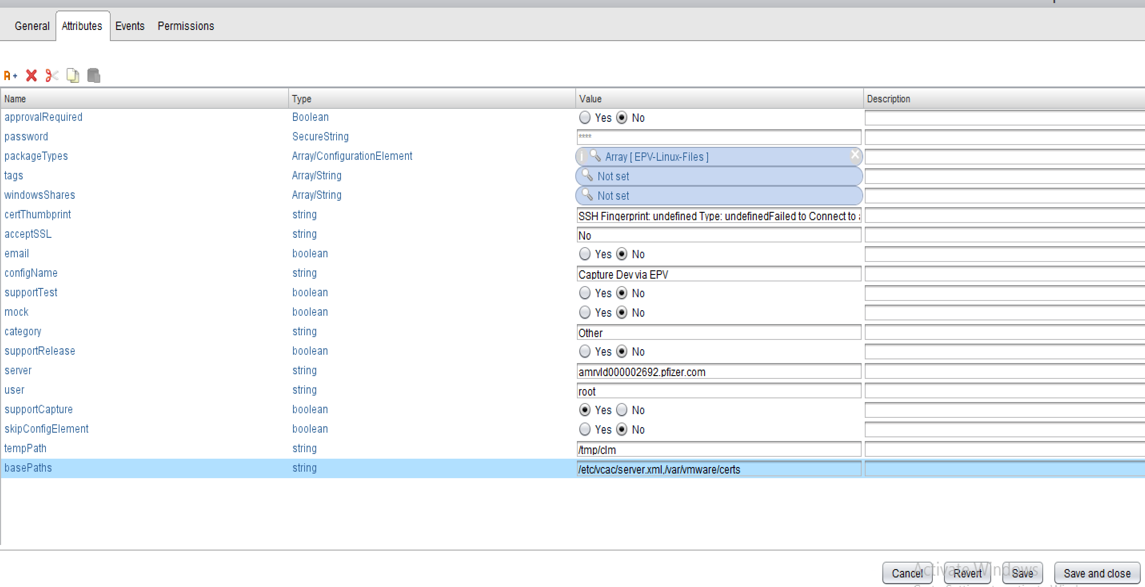


1. Insert into the basePaths variable items that are available for capture
   1. Examples
      1. File: /etc/vcac/server.xml

In this case a single file is presented for processing selection

* + 1. Directory: /var/vmware/certs

When a directory is specified, the command executed on the directory is “find <dir> -maxdepth 1 –type d”, this allows only directories of the high level to be available for capture.

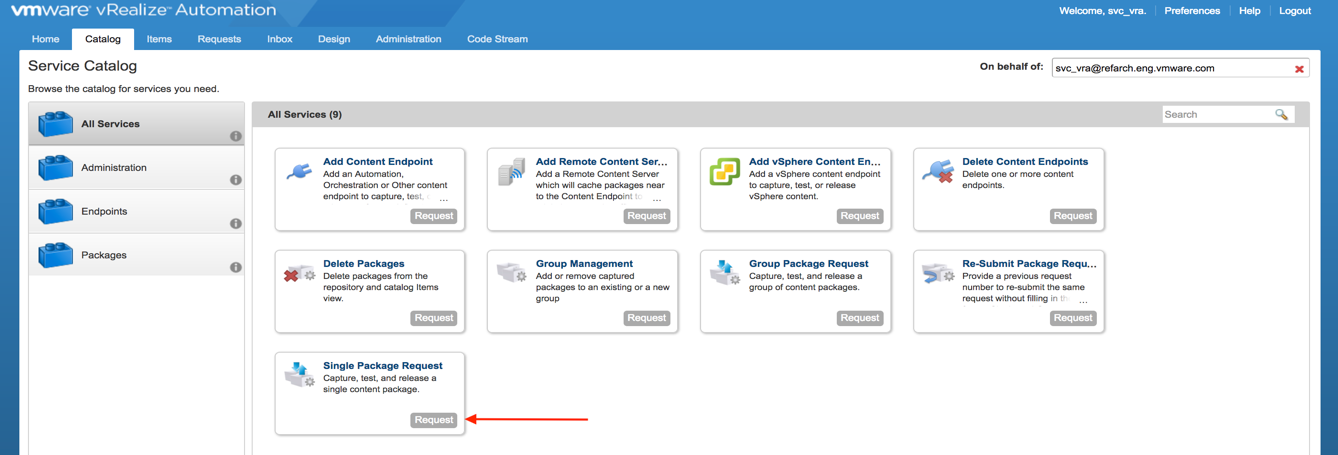


# Capture EPV-Linux-Files Example

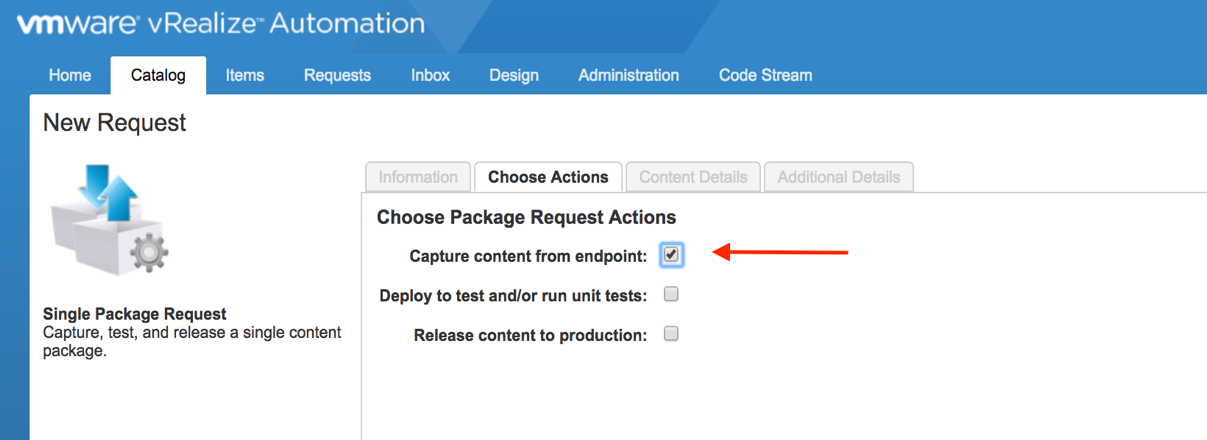
## Request a Capture

In this example a “Single Package Request” will be used to capture a File from a endpoint (server) that is under EPV control. It is assumed that the linux endpoint has been setup according to the instructions above and in EPV as a valid host.

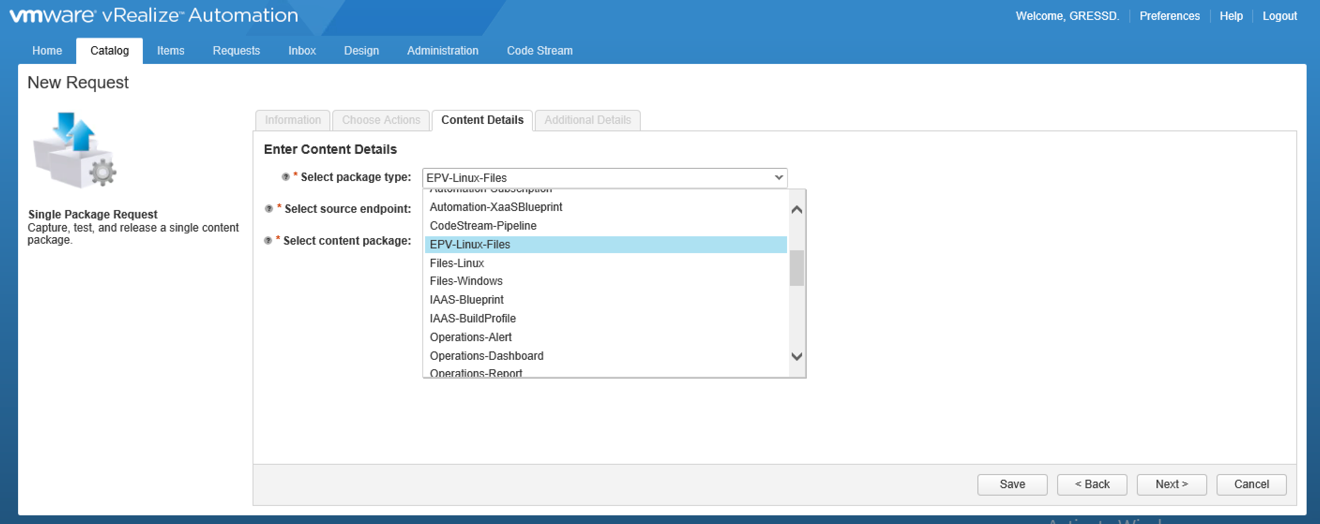
1. Login to vRA Codestream and goto the Catalog
   1. Select “Single Package Request”



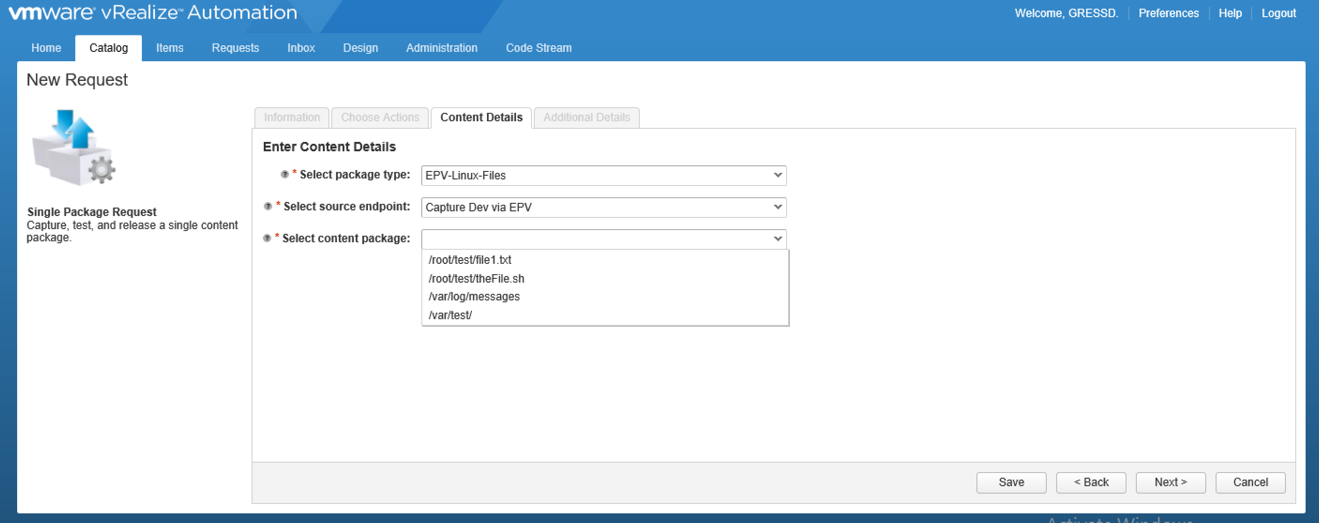
1. Click next and select “Capture content from endpoint” and click Next



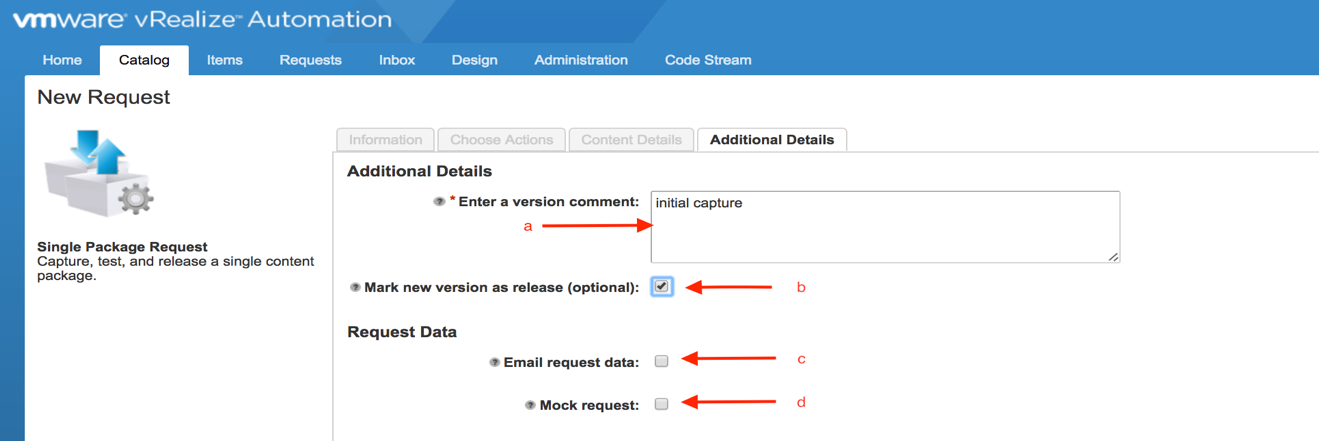
1. In the DropDown for “Select package Type’ Select “EPV-Linux-Files”



1. Select the Source endpoint ( Name was assigned when the endpoint was configured )
2. Listing of all EPV-Linux-Files will be displayed
   1. Select the file or directory
      1. Note: The files and or directories listed is controlled by the basePaths attribute added above.



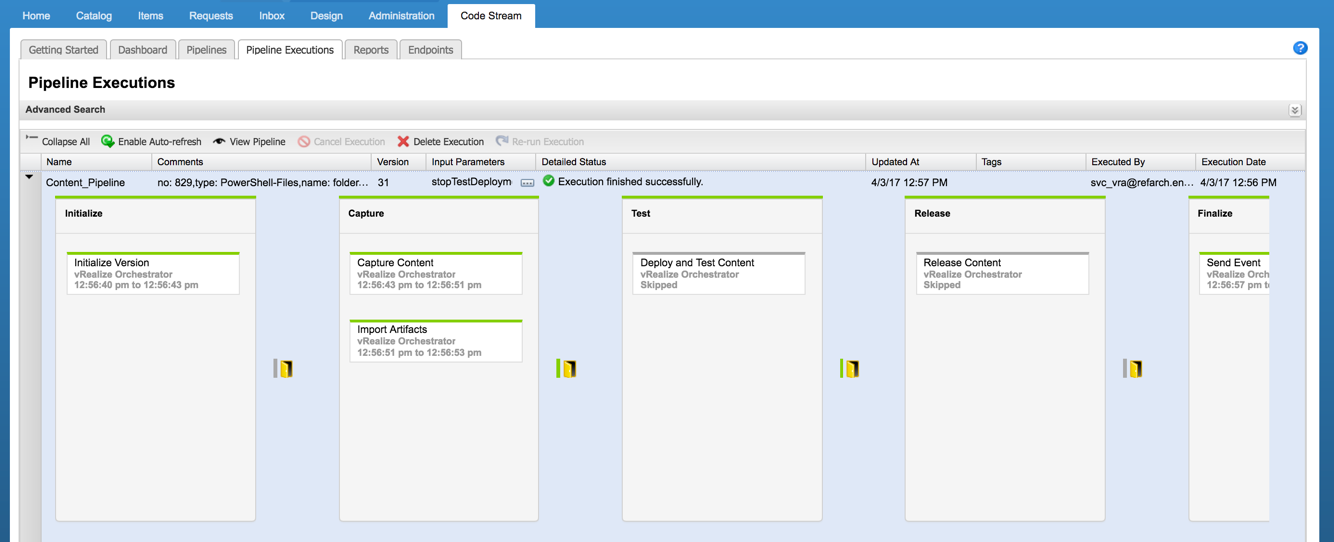
1. Hit the Next button
   1. Fill out descriptive text for the Capture
   2. Mark the capture as release (This allows moving the artifact to the next stage)
   3. Email request data : If you want a copy of the request click here, enter your email address
   4. Mock Request : Clicking this causes a “test” of the capture, leave unchecked to initiate the request



1. Hit the Submit button

## Monitor the request

1. The requests will be queued and processed.
   1. Click the CodeStram Pipeline Executions tab
   2. Enable Auto-Refresh (button) to watch execution status



1. As the request moves thru the pipeline the stages will turn Green (Success) or Red (Failure)

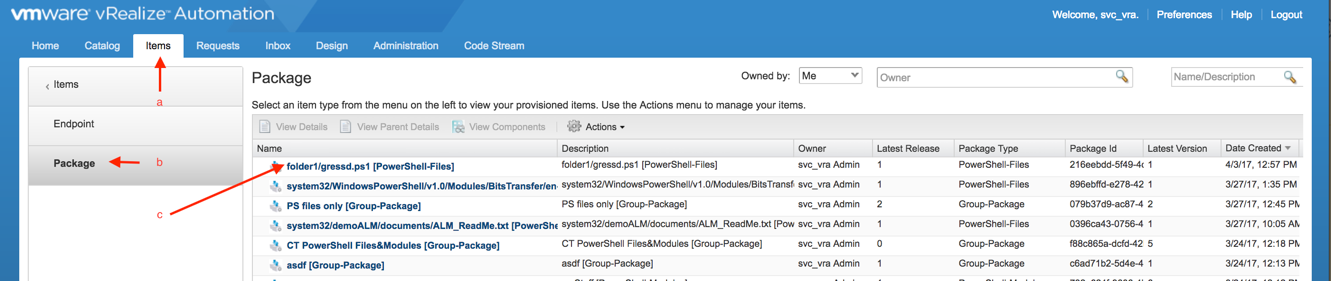
# Day 2 Operations

After capturing EPV-Linux-Files you can perform day 2 operations on those artifacts.

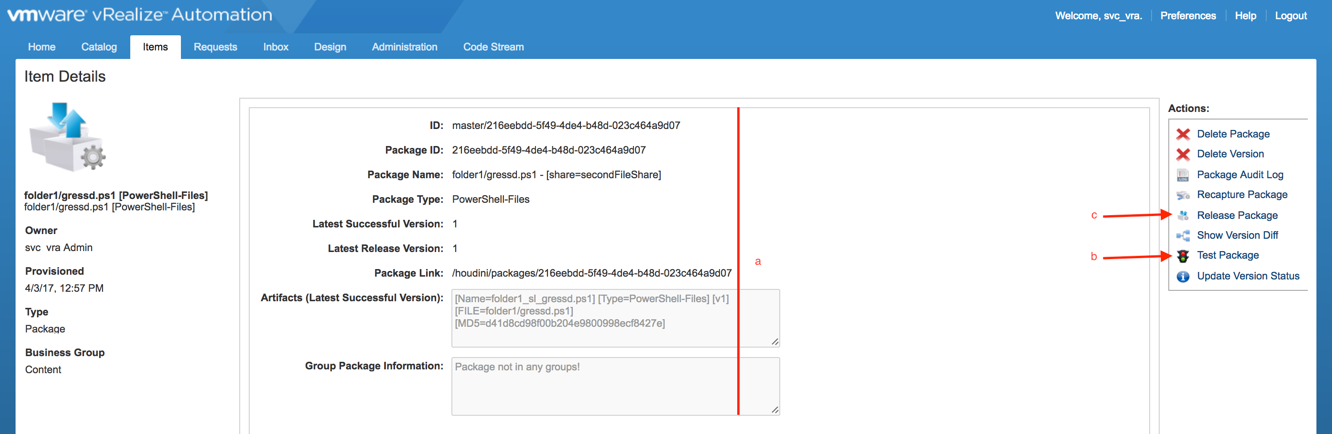
## Test Package

The “Test Package” option allows a user to send the artifact to the “Test” endpoint that is configured for “Testing”.

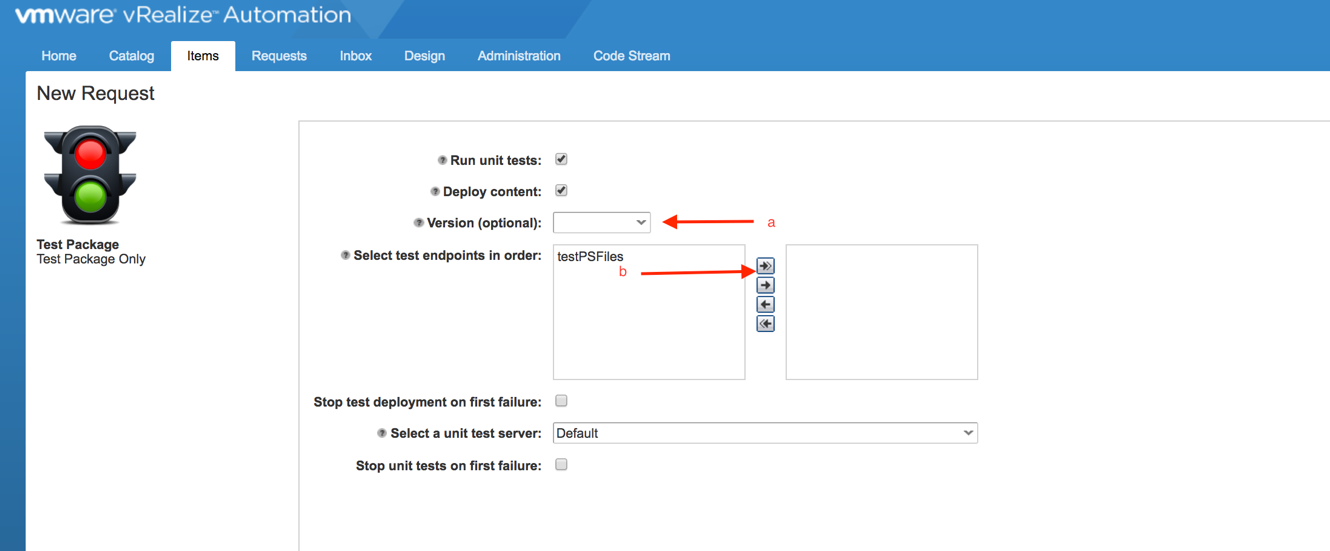
1. Select the item and initiate day 2 operations
   1. Select the Items Tab
   2. Select the Package Tab
   3. Click on the artifact link



1. Details of the artifact are now available



1. Details of the artifact
2. Test Package – Clicking this allows for the artifact to be released to the configured Testing endpoint
3. Release Package – Clicking this allows for the artifact to be released to the “Release” endpoint
4. Test endpoint and version Selection
   1. Select a version if more than one exists, the default is to use the highest version
   2. Highlight the target Test endpoint and hit the Right arrow. If multiple Testing endpoints are used, select each one of them

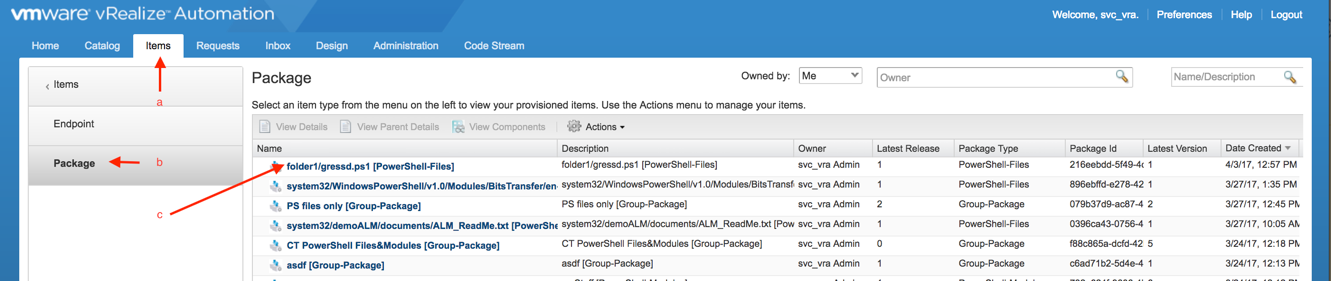


1. Hit the submit button
   1. The request is queued and you can watch the Pipeline / Pipeline executions for status update on the request

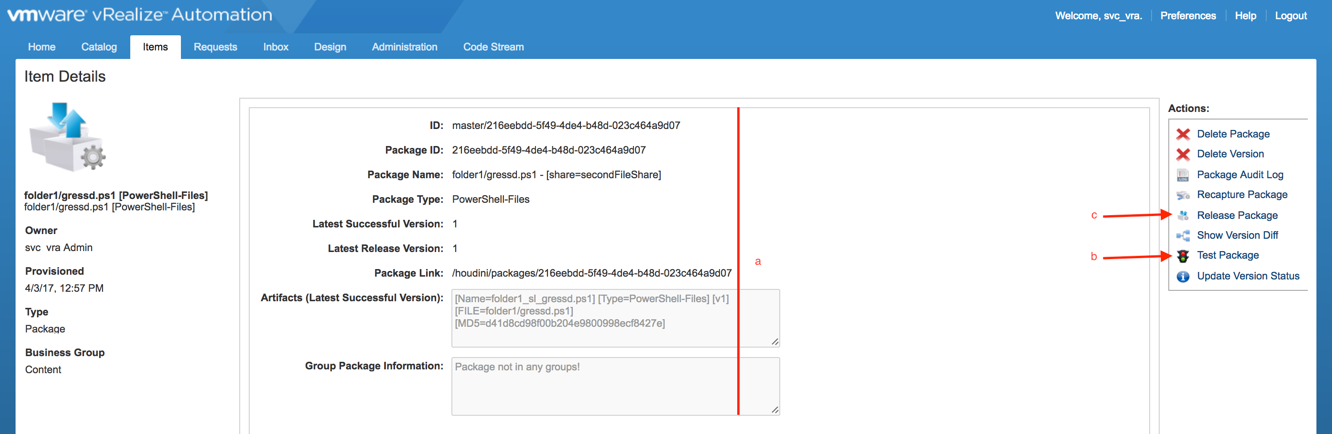
## Release Package

Releasing a package is also determined by which endpoint configurations are setup for “Release”. This is generally the 3rd phase and Production.

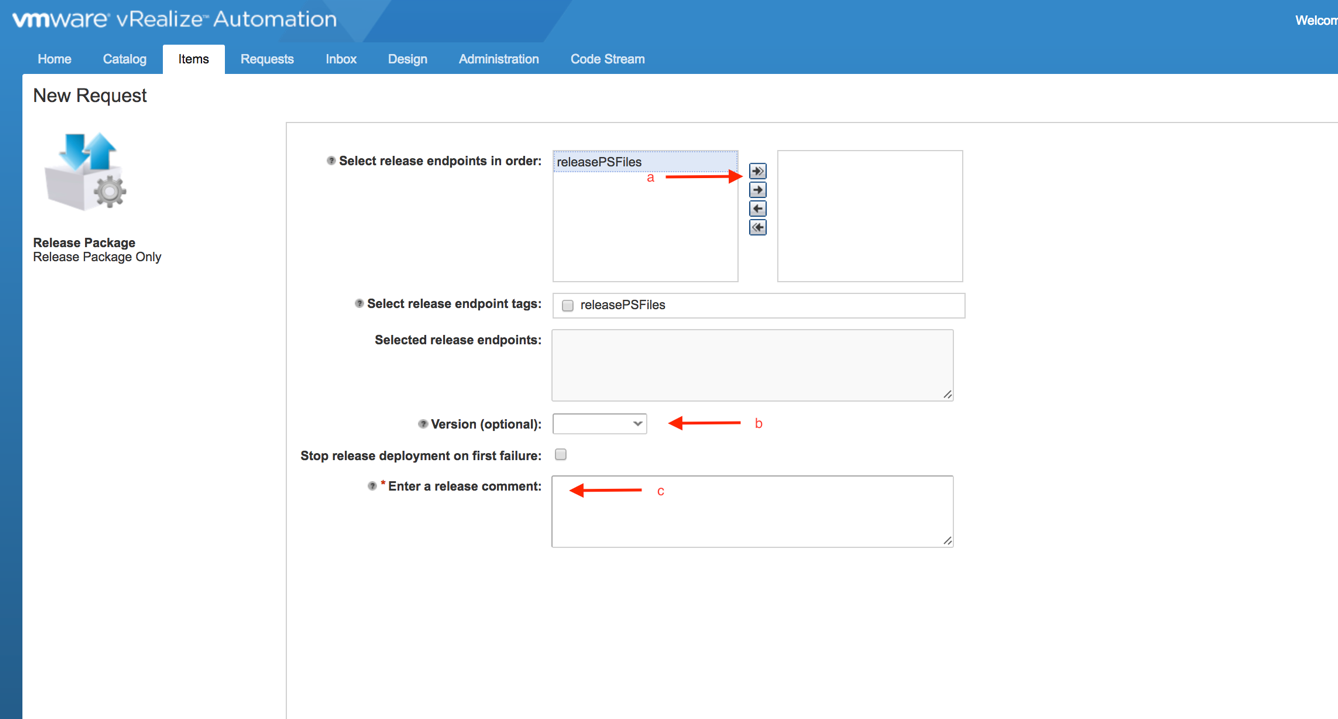
1. Select the item and initiate day 2 operations
   1. Select the Items Tab
   2. Select the Package Tab
   3. Click on the artifact link



1. Select the Release Package ©



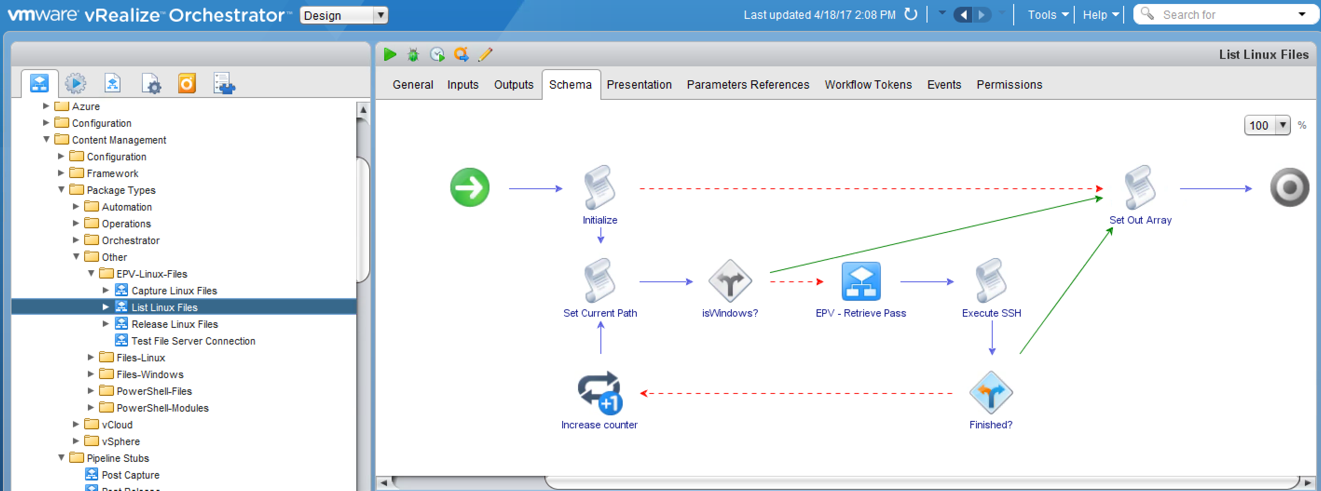
1. Fill out the New Request Release Package Form
   1. Highlight and hit right arrow for each endpoint you want the package released to
   2. Select a version (Optional), the current / highest version is selected by default
   3. Enter some descriptive comments for the release



# Schema – Flow logic

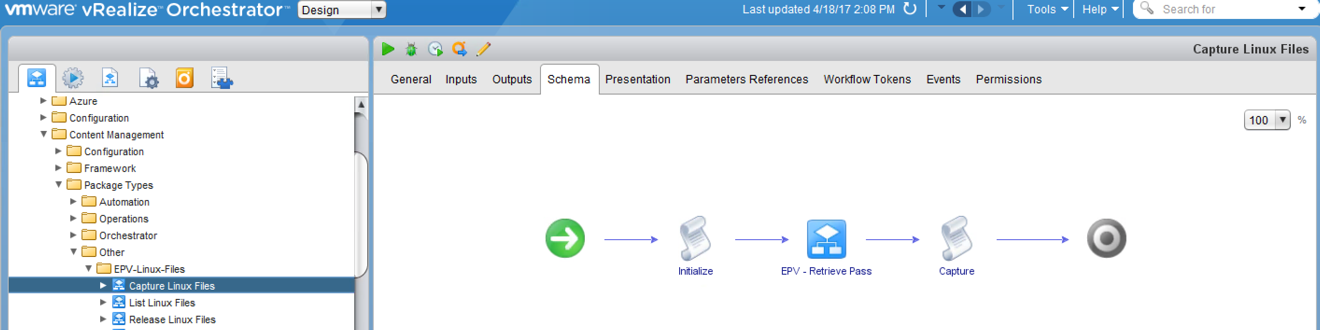
## List

* External call to obtain password via workflow : EPV – Retrieve Pass
* Loops around for all elements separated in the basePaths configuration attribute



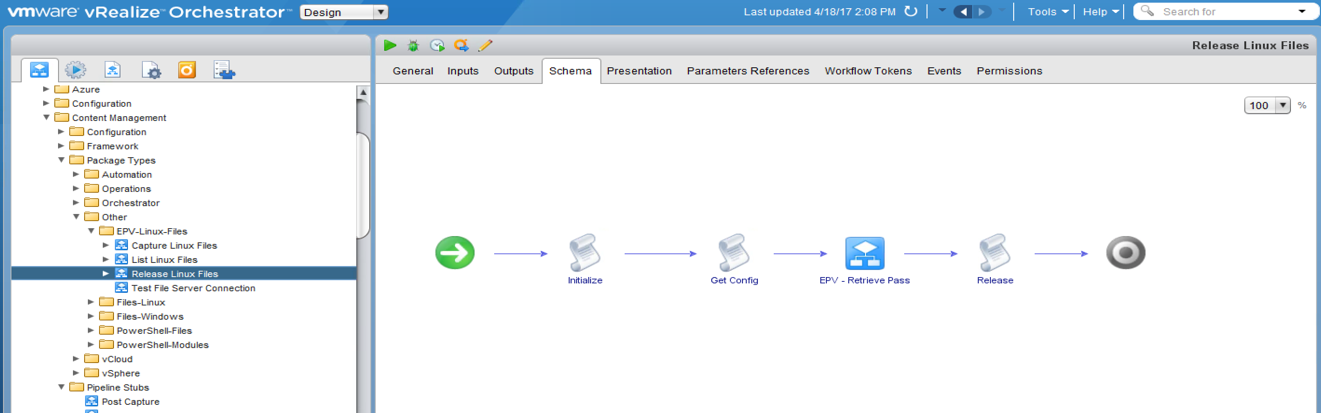
## Capture

* External Call to obtain password via workflow : EPV – Retrieve Pass



Release to endPoint

* External Call to obtain password via workflow : EPV – Retrieve Pass



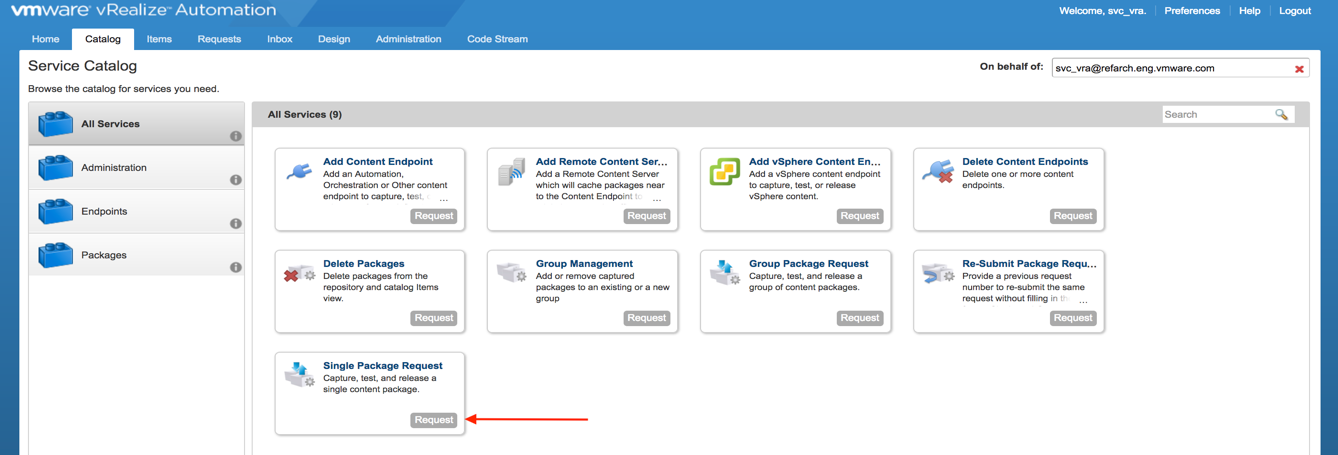
# Capture & Test

In this example a “Single Package Request” will be used to capture a File from a endpoint (server) that is under EPV control. It is assumed that the linux endpoint has been setup according to the instructions above and in EPV as a valid host.

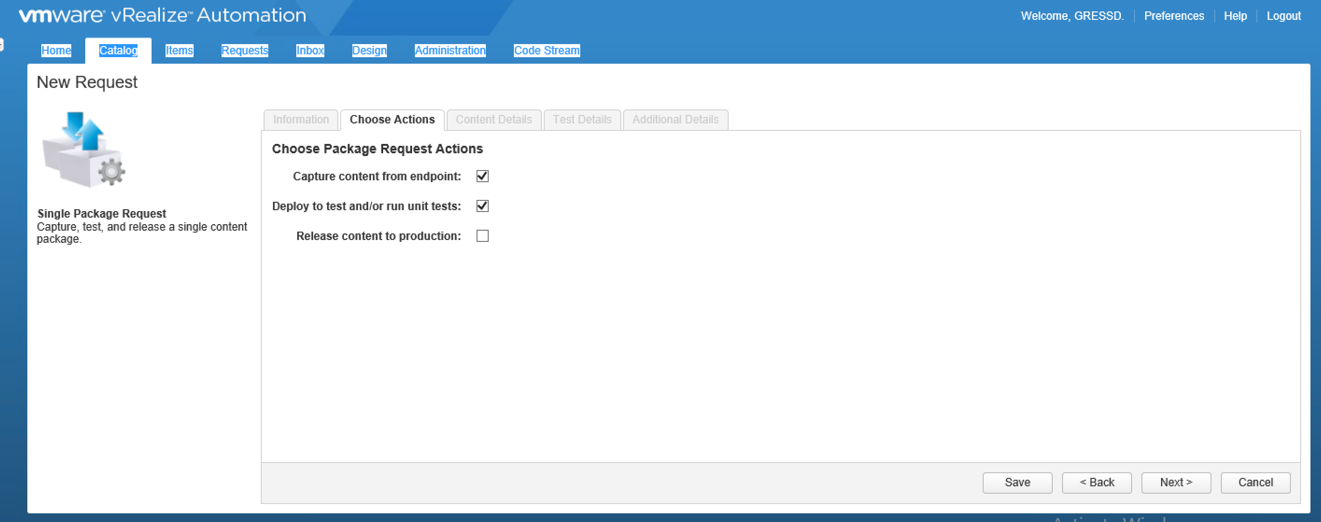
* This process captures an artifact, “orchestrator-package” in this example and releases it toward the configured “Test” endpoint or server.
* Multiple Test servers can be configured to install the artifact with or without approval
  + Pfizer has only 1 Test system to be configured with gating approvals, not shown in this example

## Initiate Pipeline

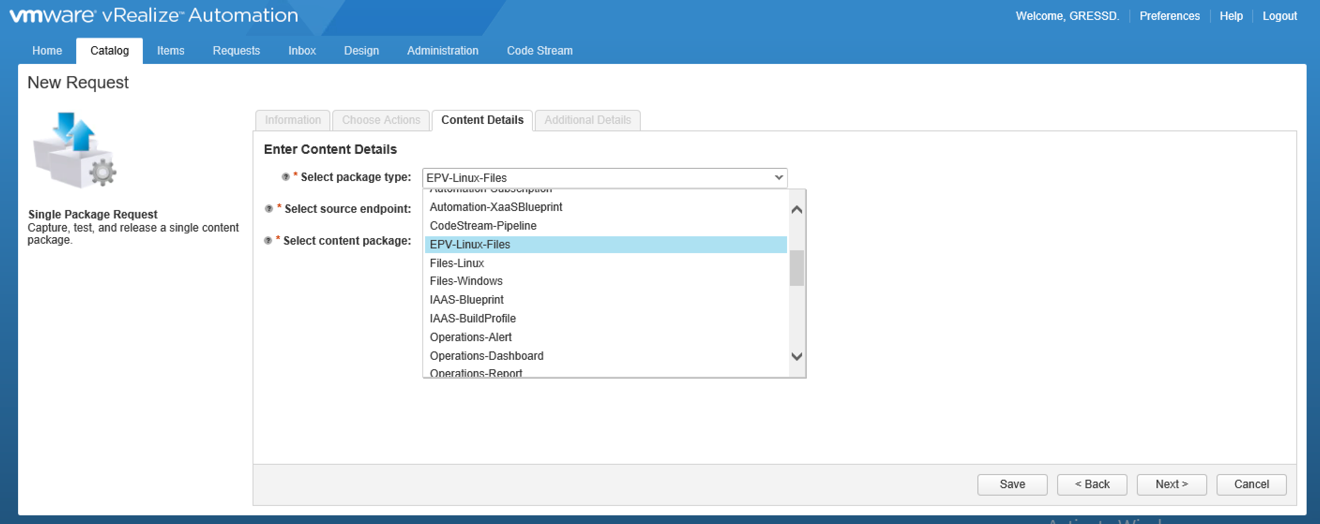
1. Login to vRA Codestream and goto the Catalog
   1. Select “Single Package Request”



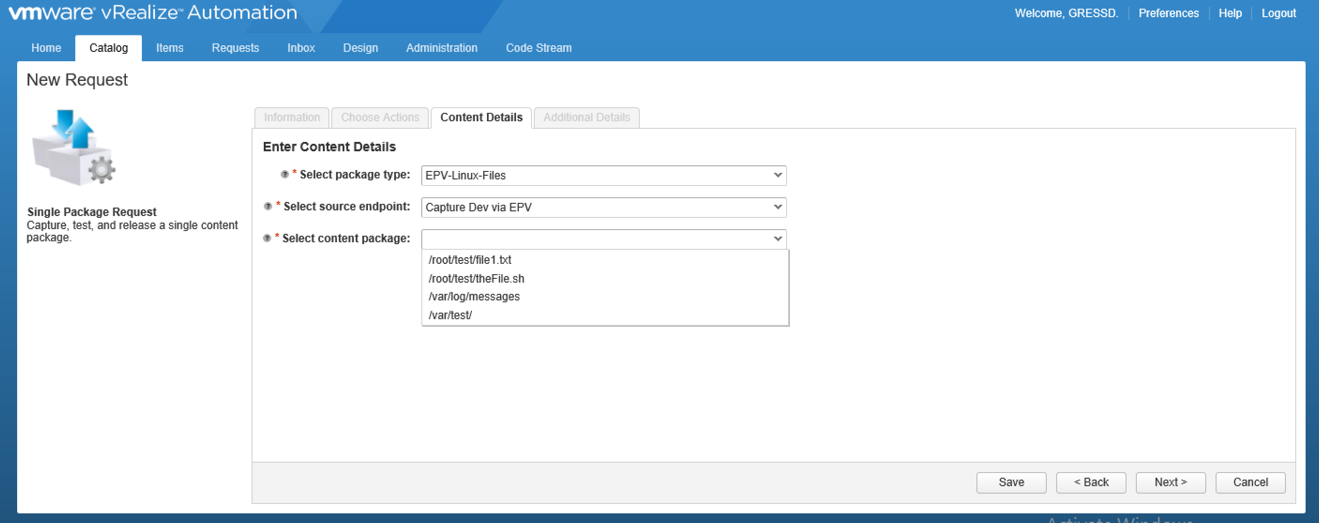
1. Click next and select “Capture content from endpoint and the Deploy to test” and click Next



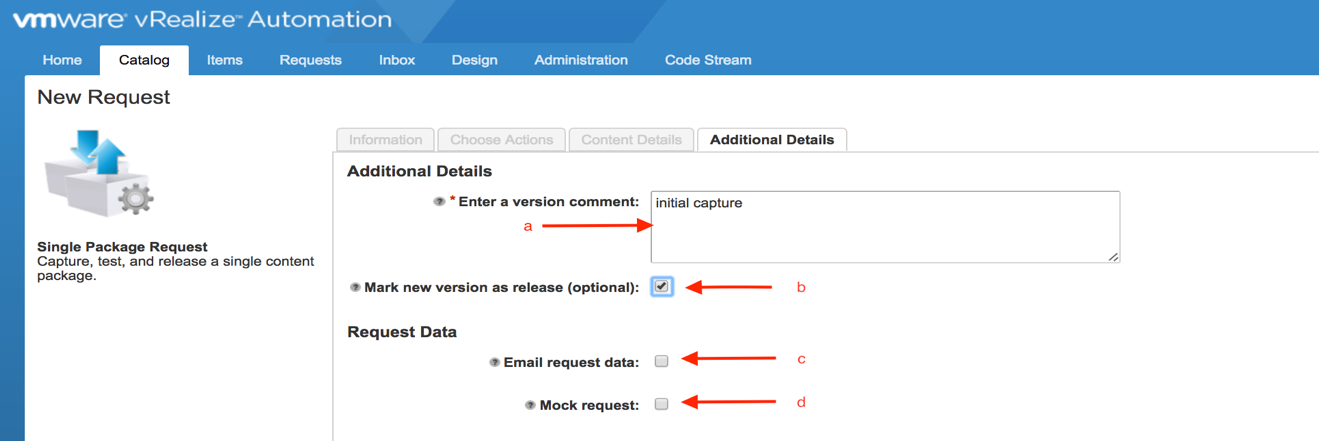
1. In the DropDown for “Select package Type’ Select “EPV-Linux-Files”



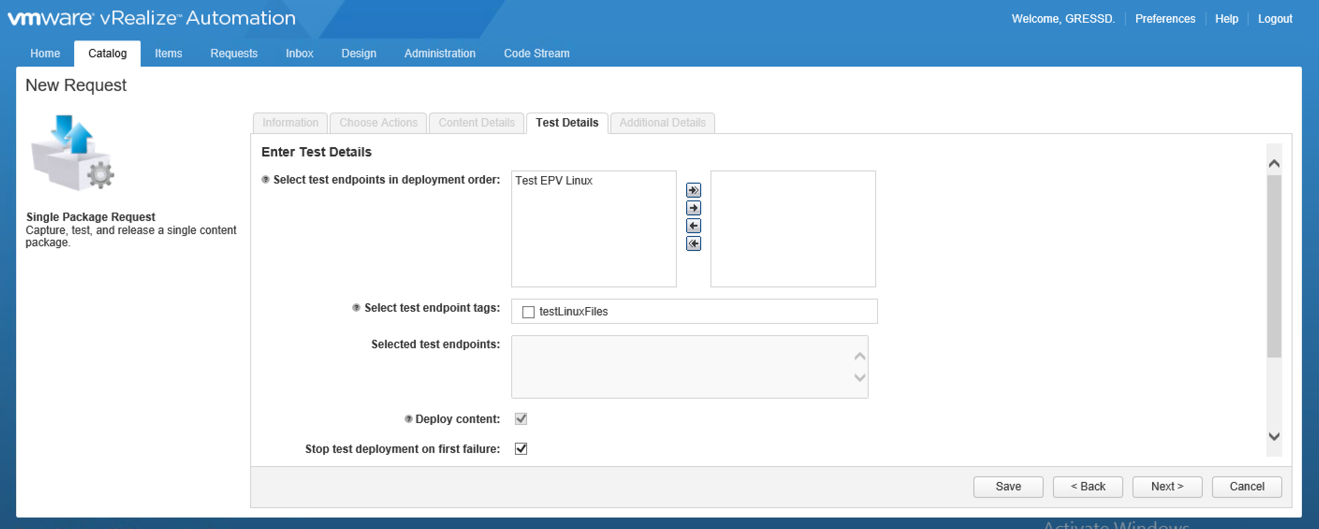
1. Select the Source endpoint ( Name was assigned when the endpoint was configured )
2. Listing of all EPV-Linux-Files will be displayed
   1. Select the file or directory
      1. Note: The files and or directories listed is controlled by the basePaths attribute added above.



1. Hit the Next button
   1. Fill out descriptive text for the Capture
   2. Mark the capture as release (This allows moving the artifact to the next stage)
   3. Email request data : If you want a copy of the request click here, enter your email address
   4. Mock Request : Clicking this causes a “test” of the capture, leave unchecked to initiate the request



1. Fill out the “Test” system form
   1. Select a “Test” endpoint
      1. If Multiple endpoints are displayed and selected, insert them into the table in order of deployment and testing.

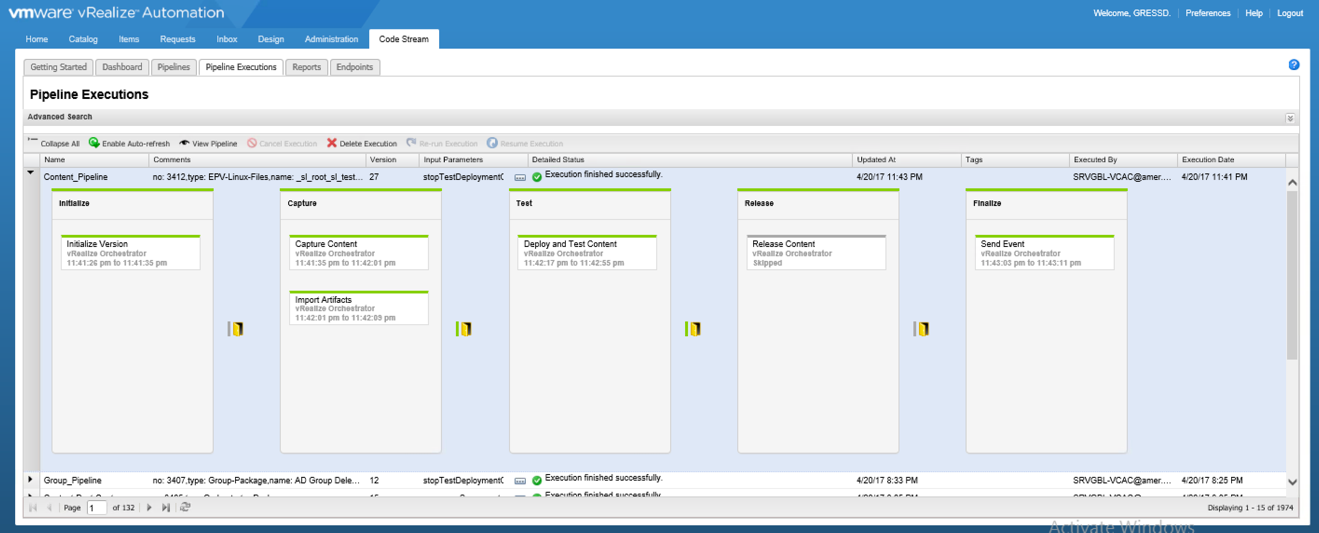


* + 1. Hit the Next button

## Monitor Results

Monitor the pipeline status via tabs

* 1. Codestream / Pipeline Status / < Selected Pipeline >
  2. Expand the pipeline with the arrow to the left
  3. Enable Auto Refresh



1. When the pipeline completes
   1. All Green
      1. processing of delivering the artifact(s) to the remote endproint have successfully completed
   2. Red
      1. An error has occurred, processing stops based on conditions setup in the pipeline tasks
      2. Click in the Red task for more information