RiteHere FS
Identify Effective File Systems Rights

RiteHere FS is a Win32 tool that allows you identify the eDirectory users with effective rights in one or more specified file system directories.

RiteHere eDir

Identify Effective eDirectory Rights

RiteHere eDir is a Win32 tool that allows you identify the eDirectory users with effective rights in one or more specified eDirectory containers.

Table of Contents

1 Overview: RiteHere

RiteHere exists in two separate flavors and with two separate executables:
- **RiteHere_E (RiteHere eDirectory)**

    RiteHere_E is a Windows 32bit application that allows you to determine who has direct or indirect trustee rights in specified eDirectory containers.

- **RiteHere_F (RiteHere Filesystem)**

    RiteHere_F is a Windows 32bit application that allows you to determine who has direct or indirect trustee rights in specified NetWare directories.

This documentation discusses usage of both: while they produce rather different specific reports, usage is very similar.
Depending on the software or software package that you have downloaded or purchased you may have one or both of these applications.

The following short documentation will indicate if comments apply to only one of the applications.

1.1 Prerequisites

- You need to use a Windows machine with the Novell Client
- You need to be logged in to the server or tree that you want to run the report on.

1.2 Quick Start

RiteHere_E is a Windows 32bit application that allows you to determine who has direct or indirect trustee rights in specified eDirectory containers.

Get results in a just few steps:
- Start RiteHere_E
- Tab "Select eDirectory Objects": Select the target tree, then the eDirectory container that you want to report on. Navigate to the appropriate container by double-clicking on one of the left side container names.
- Select one or more target objects from the center window by double-clicking it. Selected objects are listed in the bottom frame. They can be removed from the list by double-click, or they can be temporarily excluded from the report by unchecking them.
- Tab "Scan for Trustees": Press the "Scan for Effective Trustees" button and check the results

RiteHere_F is a Windows 32bit application that allows you to determine who has direct or indirect trustee rights in specified NetWare directories.

Get results in just a few steps:
- Start RiteHere_F
- Select the servers and paths that you want to check (browse to the target path, then double-click one or more paths)
- Press the "Scan for Effective Trustees" button and check the results

1.3 Purpose

NetWare and eDirectory have a complex and elaborate security system where rights can be derived from multiple sources, or restricted through filters.

Effective rights on a given **eDirectory** object are determined by several factors, e.g.:
- direct trustee assignment to the eDirectory attribute or object
- direct admin trustee assignment on a parent container of the eDirectory object
- effective rights for a given attribute may depend on trustee assignments on the "[All Attributes]" or the "[Entry Rights]"
- membership to a group or security equivalence to another object that has one of the above direct assignments
- residence in a eDirectory container that has one of the above direct assignments
- at each level in eDirectory there may be 'Inheritance Rights Filters' (IRF) – also known as 'Inheritance Rights Masks" (IRM) that may restrict trustee rights assigned on a higher level.
- Trustee rights assignments on a higher level may be expanded or restricted by further trustee assignments on lower levels.

Effective rights on a given **NetWare path/file** are determined by similar factors, e.g.:
- direct trustee assignment to the given path
- direct trustee assignment to a parent directory
- direct admin trustee assignment on the server object in eDirectory
- direct admin trustee assignment on a parent container of the server object
- membership to a group or security equivalence to another object that has one of the above direct assignments
- residence in a eDirectory container that has one of the above direct assignments
- at each level in eDirectory or in the file system, there may be 'Inheritance Rights Filters' (IRF) – also known as 'Inheritance Rights Masks" (IRM) that may restrict trustee rights assigned on a higher level.
- Trustee rights assignments on a higher level may be expanded or restricted by further trustee assignments on lower levels.

RiteHere will traverse the file system and/or eDirectory structures to identify the possible sources of the rights and calculate the resulting effective rights for users.

When displaying the list of effective trustees in the detailed output format, RiteHere will show a user's effective rights together with the reasons that cause the listed rights combination. This feature helps analyzing and – if necessary – modifying the rights structure for the tree and file system. You may switch between a summary view and a detailed view that shows your where somebody's rights stem from.

1.4 Usage: RiteHere_eDir
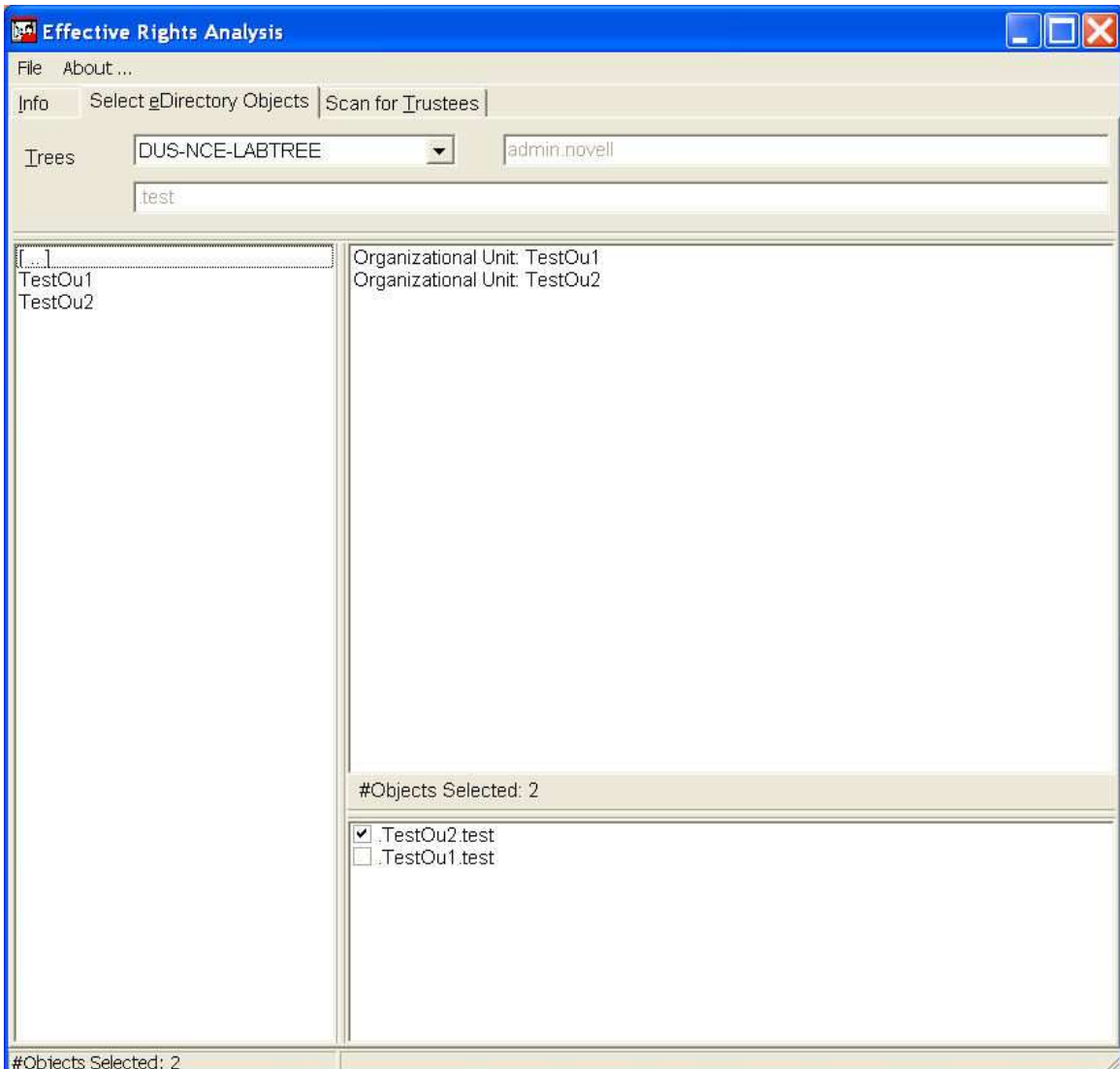
- **Tab "Select eDirectory Objects"**
  - o Select the target tree, then the eDirectory container that you want to report on. Navigate to the appropriate container by double-clicking on one of the left side container names.
  - o Select one or more target objects from the center window by double-clicking it. Selected objects are listed in the bottom frame.
  - o The selected eDirectory objects can be removed by double-click, or they can be temporarily excluded from the report by unchecking them.
  - o Unchecked objects will not be reported on. This can be useful if you regularly need to report on the same objects, but not all of them in the same report.
  - o Your selection will be saved for your next session
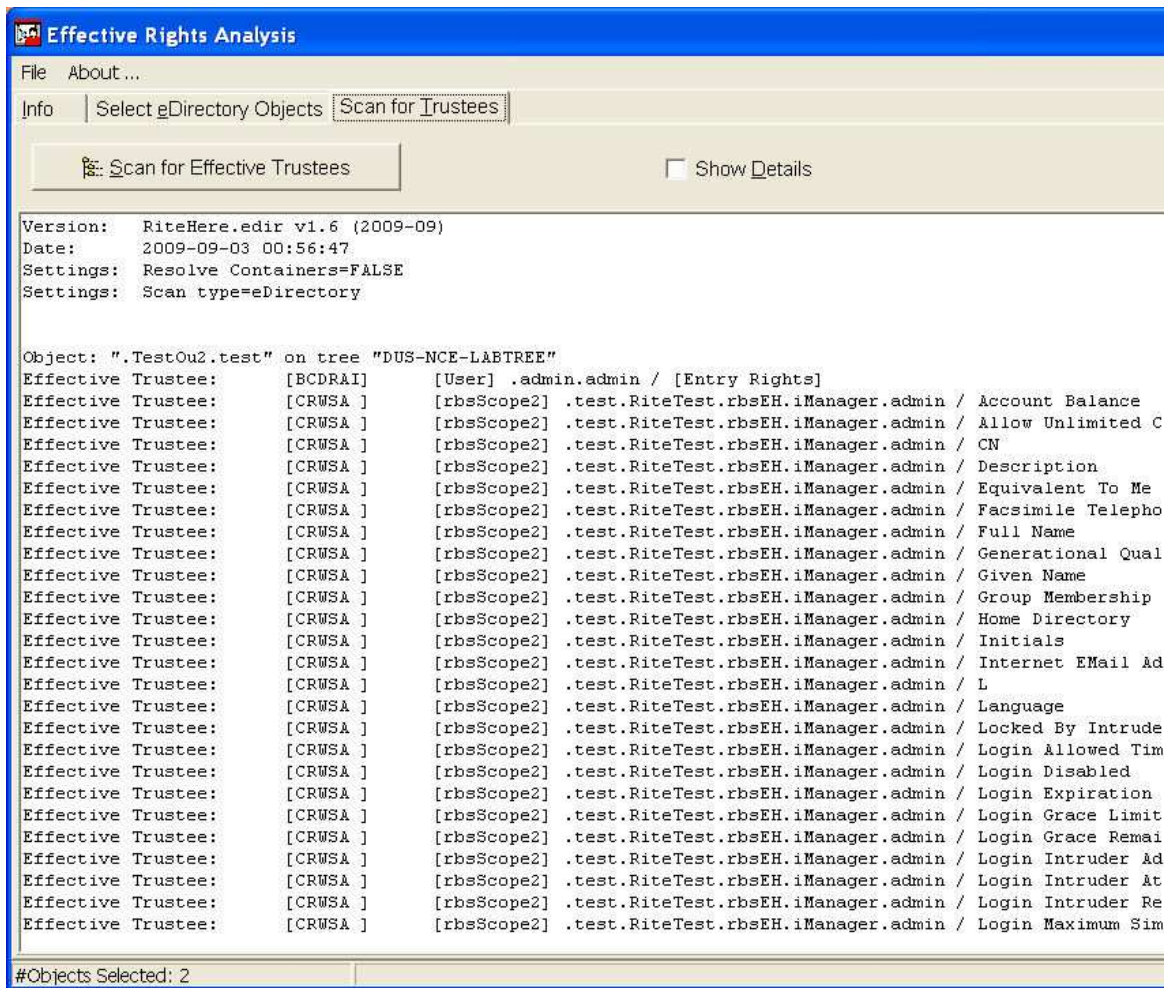
- **Tab "Scan for Trustees"**
  - Click "Scan for Effective Trustees" to start the search and let RiteHere process the data
  - The summary report will show the effective rights of the direct and indirect trustees.
  - The checkbox "Show Details" allows you to see a more detailed analysis of the scanned data. Switching between "Summary View" and "Detail View" is possible without re-scanning the paths
  - In the detailed report, you will find information on trustees on the higher/parent levels of the selected container

        o   In the detailed report, you will find information on trustees on the higher/parent levels of the selected container

```
Effective Rights Analysis
File   About ...
Info  | Select eDirectory Objects | Scan for Trustees |

    [ Scan for Effective Trustees ]                    ☑ Show Details

Version:    RiteHere.edir v1.6 (2009-09)
Date:       2009-09-03 00:56:47
Settings:   Resolve Containers=FALSE
Settings:   Scan type=eDirectory
_____


Object: ".TestOu2.test" on tree "DUS-NCE-LABTREE"

DN: [Root]
  Direct Trustee: [BCDRAI] .admin.novell / [Entry Rights]
  Direct Trustee: [B    I] [Public] / [Entry Rights]
  Direct Trustee: [BCD  I] .[root].eDirectory Administration.Role Based Service.novell / [Entry Rights]
  Direct Trustee: [CRWSA ] .[root].iPrint Management.Role Based Service.novell / [All Attributes Rights]
  Direct Trustee: [BCDRAI] .[root].iPrint Management.Role Based Service.novell / [Entry Rights]
  Direct Trustee: [BCDRAI] .admin.admin / [Entry Rights]
  Direct Trustee: [CRWSA ] .admin.admin / [All Attributes Rights]

DN: .test
  Direct Trustee: [CR   ] .test / Login Script
  Direct Trustee: [CR   ] .test / Print Job Configuration
  Direct Trustee: [CRWS ] .orgRole.novell / [All Attributes Rights]
  Direct Trustee: [BCDRAI] .orgRole.novell / [Entry Rights]
  Direct Trustee: [CRWSAI] .orgRole.novell / Given Name
  Direct Trustee: [CRWSAI] .orgRole.novell / Full Name
  Direct Trustee: [CRWSAI] .orgRole.novell / Initials
  Direct Trustee: [CRWSAI] .orgRole.novell / Surname
  Direct Trustee: [CRWSAI] .orgRole.novell / Generational Qualifier
  Direct Trustee: [CRWSAI] .orgRole.novell / CN
  Direct Trustee: [CRWSAI] .orgRole.novell / Title
  Direct Trustee: [CRWSAI] .orgRole.novell / Internet EMail Address
  Direct Trustee: [CRWSAI] .orgRole.novell / Telephone Number

#Objects Selected: 2
```
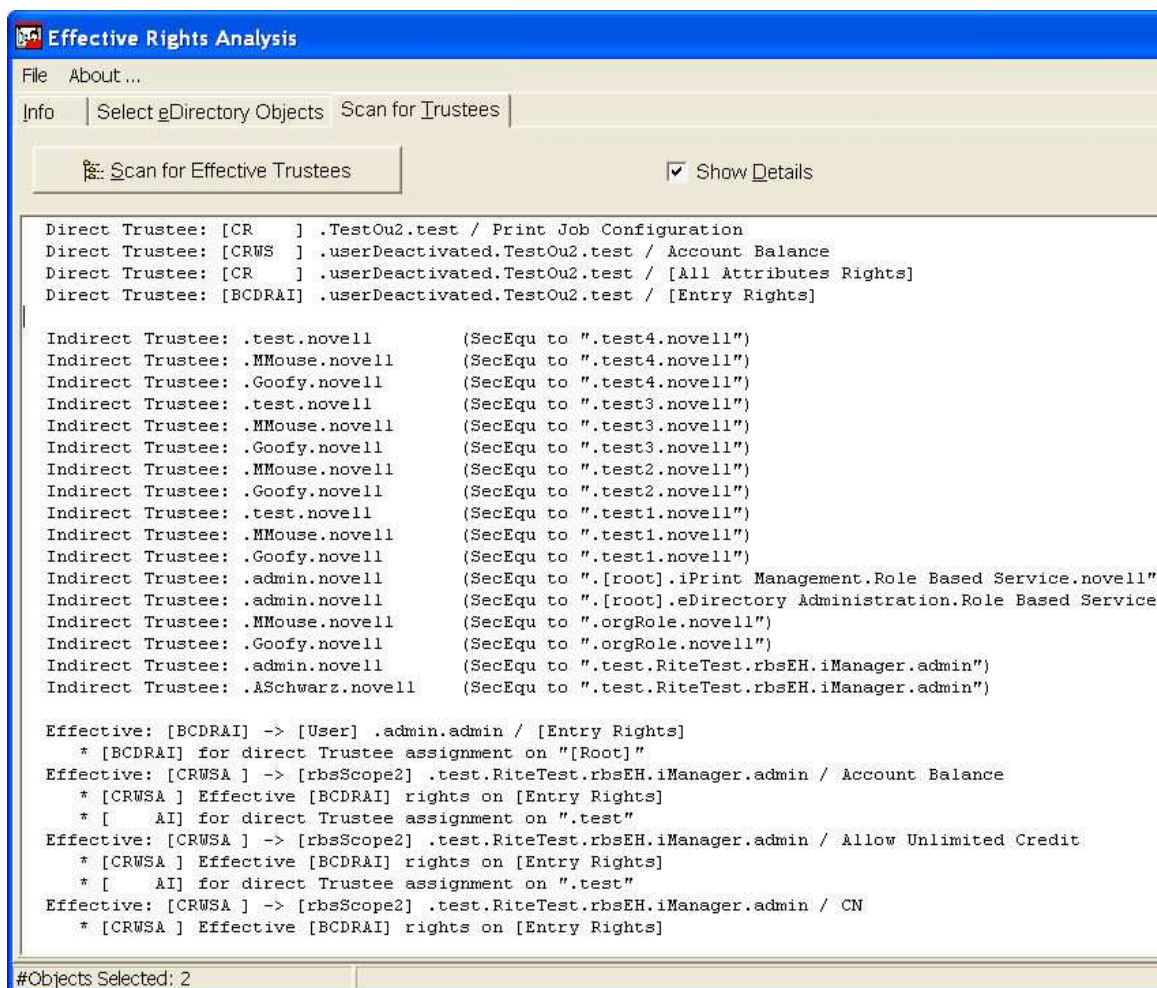
o   The detailed report will also indicate where the reported effective rights stem from. (e.g. from a direct trustee assignment, or from a security equivalence to another object.
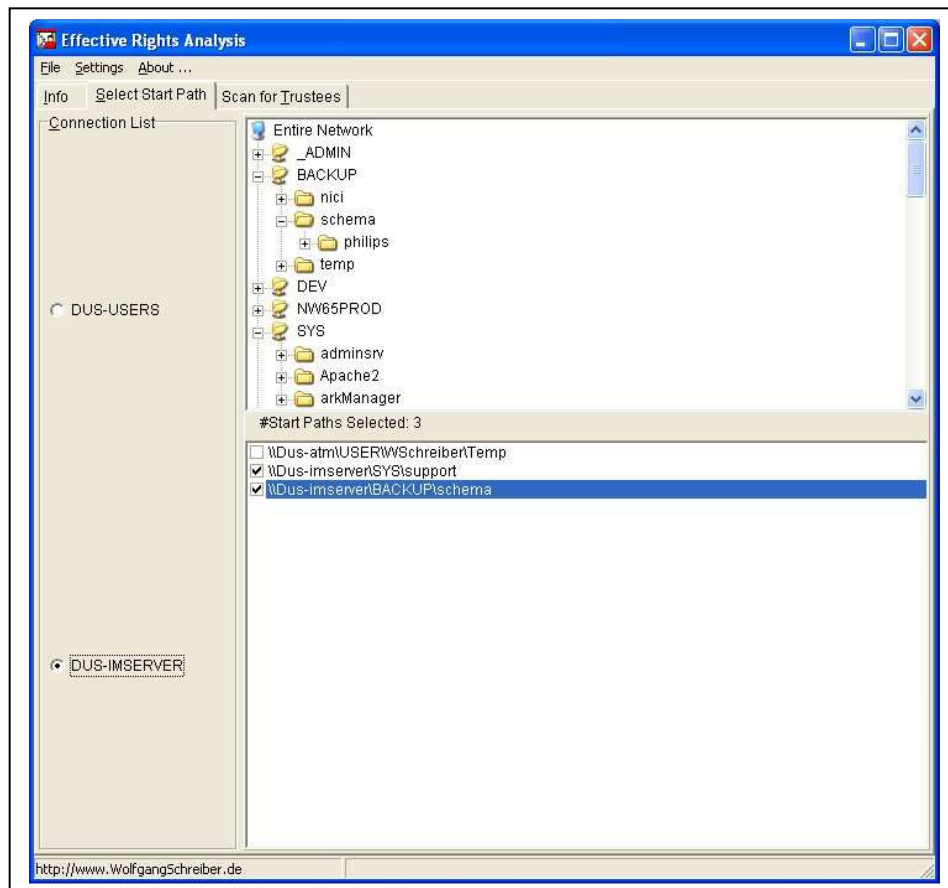

- **Getting and saving results**
    - o   The results context menu (right mouse click) offers the option to save the results as a text file (registered version).
    - o   You may save the summary output, or the details output by right-clicking in the respective window
    - o   RiteHere will filter the resulting list of effective trustees and ignore all objects that cannot login to the directory. Objects that can login are identifiable by the entry "ndsLoginProperties" in their "Object Class" attribute.

- **Settings (not available in all versions)**
    - o   "Resolve Containers"
      In some environments Organizational Units (or other container objects) may have been assigned trustee rights. All users that reside in that Organizational Unit will inherit the trustee rights granted to the container. By default, RiteHere will display the name of a container in the result list of effective trustees.
      If the menu setting "Resolve Containers" is marked, and a container is assigned trustee rights, RiteHere will resolve the container and analyze the objects that reside in the container. Use this option to decide if you want to see the eDirectory container trustees or want to analyze the users in these containers. Unchecking this option may be useful to get a more compact result set for containers where numerous users reside.

    - o   "Show Details"

If the menu setting "Show Details" is checked, the default display will be the detail view. Otherwise, the summary view is shown as default. You can always switch between detail and summary view.

1.5 Usage: RiteHere_FS

- **Tab "Select Start Path"**
    - o   Select a fileserver and a start directory (any volume or any subdirectory)
    - o   You may add one or multiple paths that are to be scanned
    - o   The path is selected by double-clicking on it, and it will be displayed in the lower window
    - o   Your selection will be saved for your next session
    - o   The checkbox indicates, if the path is included during the next scan. This allows to save a larger set of paths, and still be able to scan only a subset of these paths

- **Tab "Scan for Trustees**"
    - o   Click "Scan for Effective Trustees" to start the search and let RiteHere process the data

o The checkbox "Show Details" allows you to see a more detailed analysis of the scanned data. Switching between "Summary View" and "Detail View" is possible without re-scanning the paths

- **Getting and saving results**
  - o The results context menu (right mouse click) offers the option to save the results as a text file (registered version).
  - o You may save the summary output, or the details output by right-clicking in the respective window
  - o RiteHere will filter the resulting list of effective trustees and ignore all objects that cannot login to the directory. Objects that can login are identifiable by the entry "ndsLoginProperties" in their "Object Class" attribute.

- **Settings (not available in all versions)**
  - o "Resolve Containers"
    In some environments Organizational Units (or other container objects) may have been assigned trustee rights. All users that reside in that Organizational Unit will inherit the trustee rights granted to the container. By default, RiteHere will display the name of a container in the result list of effective trustees.
    If the menu setting "Resolve Containers" is marked, and a container is assigned trustee rights, RiteHere will resolve the container and analyze the objects that reside in the container. Use this option to decide if you want to see the eDirectory container trustees or want to analyze the users in these containers. Unchecking this option may be useful to get a more compact result set for containers where numerous users reside.
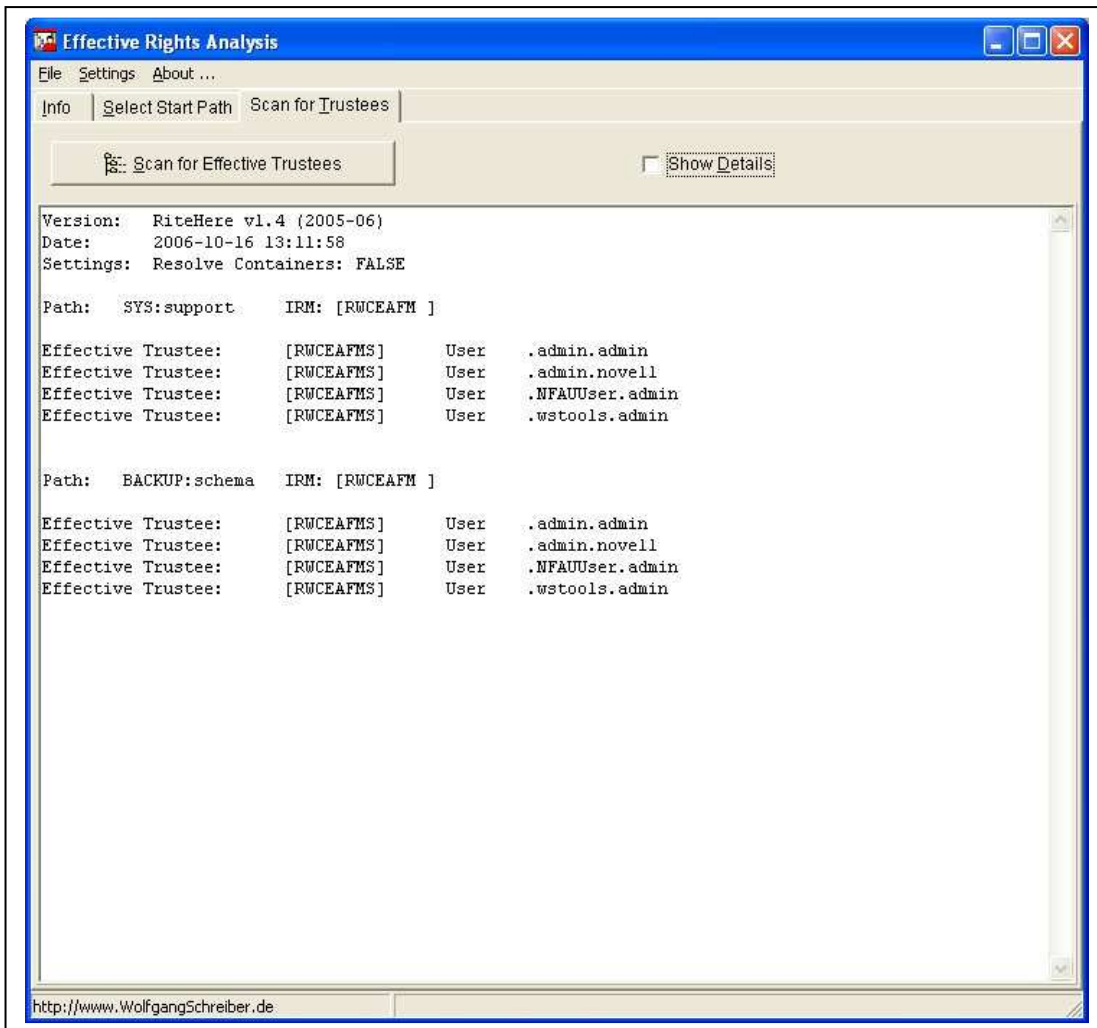
  - o "Show Details"
    If the menu setting "Show Details" is checked, the default display will be the detail view. Otherwise, the summary view is shown as default. You can always switch between detail and summary view.

1.6 Prerequisites

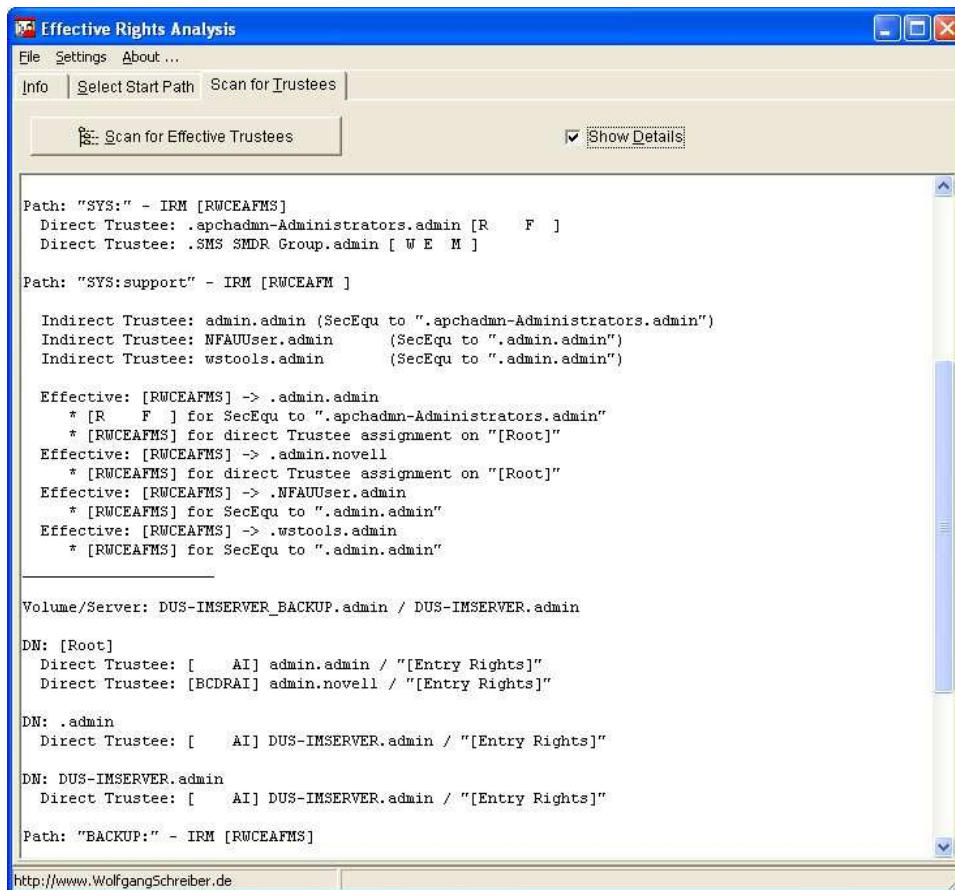- **Administrator Rights**: Since RiteHere needs to scan the file system rights as well as the eDirectory rights up to the tree root, you should have Administrator rights on both, the file system and the tree.
  If you do not have sufficient rights, RiteHere may not return all users with effective rights, since RiteHere may not be able to identify all trustee assignments or Inherited rights filters.
- **Novell Client**: RiteHere uses the Novell client software to query the NetWare file system and eDirectory.
- **Microsoft Windows (32bit):** RiteHere should run with the standard versions of Windows (e.g. Windows XP, Windows 200x, NT, Win98).
- The proxy feature of the custom version will require Windows XP or Windows 2000 with Administrator privileges on the local machine.

2 RiteHere Versions

To identify the version that you are using, see the information in the "About" option in the RiteHere menu bar.

There are two main variants of RiteHere: RiteHere_FS (Filesystem reporting) and RiteHere_eDir (eDirectory reporting). Both may come in three different flavors:

- **ShareWare version**: The unregistered free shareware versions are helpful for the occasional user: it has all standard features described in the general sections of this document. The shareware versions will allow you to scrutinize the effective user rights in the selected paths or containers, but they will not allow you to save the results. "Copy and paste" operations are restricted. These versions are available for free download at www.WolfgangSchreiber.de

- **Registered versions**: The registered versions of RiteHere have the above mentioned standard features with the addition of being able to save these results in various output formats (e.g., as text or spreadsheet). You may get the registered versions using the order form on the web site.

- **Custom versions**: Custom versions are available on request. These custom versions may provide several additional features like for example:
  o Optionally using a proxy user with different rights to check for the effective rights. This allows 'normal' users without having admin to check the rights in assigned directories.
  o Optionally formatting the trustee output to include additional information (any eDirectory attributes like 'Surname' or 'Telephone Number') about the detected trustees.
  You may get custom versions by contacting the author (wstools@WolfgangSchreiber.de) directly.

3 Changes Log

- 2004/02 Initial version
- 2005/03 Switch between summary view and details view
- 2005/04 Addition of customizable user attributes to the detail result view
- 2005/05 Addition of explanations why the users have the listed effective rights
- 2006   Handling extended characters
- 2009/09 Minor bug fixes; performance tuning

4 Additional Information

4.1 RiteHere configuration files

There are 3 RiteHere configuration files that store the settings and determine the application behavior. None of the 3 files is essential for the operation of RiteHere, and if necessary, the configuration files will be auto-created if needed.
This reference chapter is for the advanced user who wants to understand the Ritehere settings or change settings without starting the RiteHere GUI.

- ## Shared RiteHere.ini (in the same path as RiteHere.exe)

  This is a common setup file shared by all users. The file contains these main sections

  [ProgInfo]
  Licensing information is stored in encrypted format. If not present, RiteHere will run in Shareware mode

  [Notes]
  Licensing information is shown in readable format. Optional.

  [Attributes]
  See details above – information about attribute display format. If not present, no user attributes will be displayed.

- ## Local RiteHere.ini (in the Windows path, e.g. C:\Windows)

  This is an individual, user specific setup file keeping information about preferred options, form position and settings.

- ## Shared RiteHere.Prx (in the same path as RiteHere.exe)

  This file will only be present in the custom RiteHere version which implements the proxy user concept to access the directory.

4.2 Additional Features

- ## Resolve Containers

  In some environments Organizational Units (or other container objects) may have been assigned trustee rights on other objects. All users that reside in that Organizational Unit will inherit the trustee rights granted to the container. By default, RiteHere will display the name of a container in the result list of effective trustees. If the option "Resolve Containers" is set to "on", RiteHere resolves the contents of the container and displays the names of all users in the container. The additional overhead of determining the respective users may result in slower performance, dependant on the number of users in the container.

- **Output Filtering**

    RiteHere will filter the resulting list of effective trustees and ignore all objects that cannot login to the directory – mainly users. Objects that can login are identifiable by the entry "ndsLoginProperties" in their "Object Class" attribute.

- **Specifying the source of the rights**

    Object may receive effective rights through a variety of factors:
    - They may have direct trustee assignments
    - Indirect trustee assignments through security equivalence to a direct trustee (e.g., by group membership)
    - Indirect trustee assignments through their residence in a container that has a direct trustee assignment

    In addition to these direct or indirect trustee assignments, rights may be modified or revoked through
    - Different Trustee assignments on a lower level
    - Inherited rights filters on a path, container or leaf object

    When displaying the list of effective trustees in the detailed output format, RiteHere will show a user's effective rights together with the reasons that cause the listed rights combination. This feature helps analyzing and – if necessary – modifying the rights structure for the tree and file system.

5 Addendum: RiteHere – Custom Version

This documentation is based on – and extends – the standard RiteHere documentation. It serves as an addendum covering the additional or varying features of the custom version of RiteHere.exe

5.1 Customizing trustee output details

- ### Purpose

    To display additional trustee detail in custom format, use the [Attributes] section in RiteHere.ini These additional data will not be visible in the 'raw' detail trustee information screen, but only in the summary screen.

    Example

    ```
    [Attributes]
    AttrCount=5
    Attr_01=Given Name
    AttrDefault_01=
    Attr_02=Surname
    AttrDefault_02=
    Attr_03=OU
    AttrDefault_03=n/a
    Attr_04=Telephone Number
    AttrDefault_04=+44-1234-0
    Attr_05=Company
    AttrDefault_05=n/a
    AttrFormat="#TAB#TAB#Attr_01 #Attr_02#CRLF#TAB#TAB(Tel:
    #Attr_03)#TAB[Company: #Attr_05 / #Attr_04]"
    ```

- ### INI Field Explanation

    "**AttrCount**" (Default: 0) the number of additional attributes to read from the trustee object
    "**Attr_xx**" (Default: '') represents the eDirectory name of an attribute to read.
        It is recommended to only use single-valued attributes, to avoid mangling up the output.
        'xx' should be a two-digit number between '01' and the value entered in 'AttrCount'
    "**AttrDefault_xx**" (Default: '') represents the default attribute value that is used,
        if the respective eDirectory attribute is not set or is unavailable.
        'xx' should be a two-digit number between '01' and the value entered in 'AttrCount'
    "**AttrFormat**" represents the display format for the retrieved attributes.
        You can use the placeholders '#Attr_xx' for the respective attribute values.
        '#TAB' is the placeholder for a tab character
        '#CRLF' is the placeholder for a line break
        Other characters will be displayed unchanged
        Note: For exporting output into Excel, tabs will act as cell separators

5.2 Proxy Settings

- **Purpose**

The custom version of RiteHere version includes the option to use a proxy user to retrieve the requested information from eDirectory. This may be useful if you want to allow users to run RiteHere reports on effective rights, but do not want to grant them the required admin rights that they need to identify all trustees.

A proxy user is an eDirectory identity that is used to access information or resources on the directory. Typically an identity with more rights than the current user will be used as proxy identity.
Since RiteHere relies on reading trustee information on the file system and/or reading ACLs on eDirectory, a typical proxy identity will have full or partial admin rights on the file system and/or eDirectory.

By default, RiteHere uses the identity of the logged-in user and does not use any proxy users. However, you optionally may define one proxy identity per tree.

- **Defining proxy settings**

The identities of the proxy users are stored in the file "RiteHere.prx" which needs to reside in the same path as the executable (RiteHere.exe). You can use RiteHere to create and edit the file "RiteHere.prx":

- Login to the trees that you wish to create proxy settings for.

- Run the application with the 'setup' option: "RiteHere /SETUP" - RiteHere.exe should reside on a local or network path where you have create/write privileges.
  - o You will see a tab with the title "Proxy Settings" and a list of trees that you are connected to
  - o For each tree that you want to create a proxy identity for, enter a valid eDirectory name in the format ".proxyname.orgunit.org" (e.g., ".admin.novell").
  - o For each proxy user, enter the password. Typically you should enter the password in UPPERCASE characters (case sensitivity depends on the eDirectory/client versions)
  - o You can use the test button to verify the entered combination of tree name, proxy name, and proxy password
  - o Mark the checkbox "Use proxy"
  - o Close the application

- The file "RiteHere.prx" should now have been created in the executable path. For each tree that you have selected, it will contain sections like …

```
[THISTREE]
User=.proxy.test.admin
Pwd=0318047F73726A7A5E5811290D18021531191126
UseProxy=1

[THATTREE]
User=.admin.services.admin
Pwd=153F093A370A23316E52407B6249595C47415353
UseProxy=1
```

- While the "Pwd" setting is encrypted and should not be edited; the other lines can be edited as needed in the *.prx file or by re-running "RiteHere /setup".

- **Troubleshooting Proxy Usage**
  - Before distributing proxy settings, you should test the functionality of the proxy settings
  - You can use the test button to verify the entered combination of tree name, proxy name, and proxy password.
  - If you change the proxy user account and/or password in eDirectory, you will also need to update the proxy settings.
  - If proxy settings are not correct (e.g., a non-existent user name or incorrect password), the program will run with the identity of the default (logged in) user.
  - If proxy settings do no work as expected, you may run RiteHere with the "debug" parameter: "RiteHere /DEBUG" to additional troubleshooting information and error messages – this option can also be combined with the "setup" option: "RiteHere /SETUP /DEBUG"

- **Distributing proxy settings**
  - Edit the file "RiteHere.prx" as needed and copy it to the path that contains the "RiteHere.exe" application.
  - In some situations it may seem appropriate to allow users to view the proxy identity. In this case you can add this section to the *.prx file:
        [SETTINGS]
        View=1
  - To allow users to change the proxy settings, use "View=2"
  - Users will never be able to see the assigned proxy passwords, though.

- **Requirements**

  To use the proxy functionality, the current user must be a local workstation administrator
  The application was designed for Windows XP workstations with the current Microsoft service packs (as of spring 2005) and the current Novell client software (Novell Client 4.9x) installed.

5.3 *Custom Information in the shared RiteHere configuration files*

Some additional custom configuration information may be contained in the RiteHere configuration files that were mentioned above.

- **Shared RiteHere.ini (in the same path as RiteHere.exe)**

  Example file:

  [Settings]
  Resolve Containers=0
  Allow Detail=0
  Allow Resolve=0

- ## **INI Field Explanation**

  - **Optional: "**Allow Detail" (Section '[Settings]')
    - o "Allow Detail" defines whether users can select the detail view of the trustee analysis.
    - o "0": Detail View switch is not allowed / "1": Detail View switch is allowed
    - o Default is "0"

  - **Optional: "**Allow Resolve" (Section '[Settings]')
    - o "Allow Resolve" defines whether users can change the setting to resolve containers for the trustee analysis.
    - o "0": Resolve switch is not allowed / "1": Resolve switch is allowed
    - o Default is "0"

  - **Optional: "**Resolve Containers" (Section '[Settings]')
    - o "Resolve Containers" defines the initial setting of the flag to resolve containers for the trustee analysis.
    - o "0": Do not resolve containers / "1": Resolve containers
    - o Default is "0"
    - o

- ## **Shared RiteHere.prx (in the same path as RiteHere.exe)**

  This is a common setup file shared by all users. The file contains one section for every tree that you have defined a proxy user for. See details above. User names will be in clear text format, passwords will be encrypted.

  Example file:

  [Settings]
  View Proxy Settings =0
  Check Dir Rights=1

  [MYTREE]
  User=user name
  Pwd=encrypted password
  UseProxy=enable or disable proxy user for this tree

  [TESTTREE]
  User=user name
  Pwd=encrypted password
  UseProxy=enable or disable proxy user for this tree

- ## **INI Field Explanation**

  - **Optional: "**Check Dir Rights" (Section '[Settings]')
    - o "Check Dir Rights" defines if – when using proxies – the current user needs at least admin privileges or is stored as owner on the selected path.
    - o "0": Do not check rights / "1": Check rights
    - o Default is "0"

  - **Optional: "**View Proxy Settings" (Section '[Settings]')
    - o "View Proxy Settings" defines if – when using proxies – the current user is allowed to view and/or modify the current proxy settings (user name/encrypted password
    - o "0": Hide proxy settings / "1": Show proxy settings (RO) / "2": Show proxy settings (RW)
    - o Default is "0"

  - **Optional: "**UseProxy" (Section '[<treename>]')

- o "UseProxy" defines if a proxy user will be used for the tree specified as section name
- o "0": Do not use proxy / "1": Use proxy
- o Default is "0"

- **Optional: "**User" (Section '[<treename>]')
  - o "User" defines the eDirectory name of the proxy user – required if "UseProxy" is enabled

- **Optional: "**Pwd" (Section '[<treename>]')
  - o "Pwd" contains the encrypted password of the proxy user – required if "UseProxy" is enabled