

---

*WANdisco*  
*Subversion Access Control*  
*Administration Guide*

---



## Revision History

REVISION	DATE
1.0	October 2008

This material is confidential to WANdisco and may not be disclosed in whole or in part to any third party nor used in any manner whatsoever other than for the purposes expressly consented to by WANdisco in writing.

This material is also copyright protected and may not be reproduced, stored in a retrieval system or transmitted in any form or by any means in whole or in part without the express written consent of WANdisco.

# Contents

1	Introduction	1
1.1	WANdisco Access Control Concepts and Terms	2
1.1.1	Users, Roles, and Groups	2
1.1.2	Access Control Lists	3
1.1.3	Reports	3
1.1.4	Sequence For Implementing Access Control	4
1.1.5	Perl Style Regular Expressions and Syntax	4
1.2	Terms	5
2	Recommended Deployment Practices	6
2.1	Access Control Administrator Pre-requisites	6
2.2	Physical Environment	6
2.3	WANdisco and Subversion Password Files	7
2.4	Firewalls and Virus Scanners	7
2.4.1	Firewalls	7
2.4.2	Virus Scanners	7
2.5	Deployment Checklist	8
2.6	New Subversion Repository	10
2.7	Using HTTP with Apache	10
3	Installation	13
3.1	First Time Installation	13
3.1.1	Security Agent Settings	15
3.1.2	Subversion Settings	15
3.1.3	Checklist	16
3.2	Installing Upgrades	18
3.2.1	Disconnect SVN Users	18
3.2.2	Export WANdisco Data	18
3.2.3	Stop WANdisco	18
3.2.4	Preparing the Install Node	19
3.2.5	WANdisco-supplied .jar and License Key Files	19
3.2.6	Running the Install Program	19
3.2.7	Importing WANdisco Data	20
3.2.8	Verifying Installation	20

# Contents, cont'd

4	Using the Admin Console .....	21
4.1	Starting the Admin Console .....	21
4.2	The Security Page .....	22
4.3	The System Page .....	24
4.4	The Proxy Page .....	25
4.5	The Reports Page .....	26
5	About Users, Roles and Groups .....	27
5.1	Managing Roles and Permissions .....	27
5.1.1	Special Permissions .....	29
5.1.2	Creating New Roles .....	30
5.1.3	Editing Existing Roles .....	30
5.1.4	Deleting Roles .....	30
5.2	Managing Groups .....	31
5.2.1	Creating New Groups .....	32
5.2.1.1	Defining Group Rules .....	33
5.2.2	Creating a Sub-Group .....	33
5.2.3	Deleting a Group .....	34
5.2.4	Assigning Users to a Group .....	34
5.2.5	Deleting Users from a Group .....	35
5.2.6	Who Is In a Group? .....	36
5.2.7	Importing Existing Groups .....	36
5.3	Managing Users .....	37
5.3.1	Creating or Removing Users .....	37
5.3.2	Assigning Users to a Sub-Group .....	37
5.3.3	Listing and Searching for Users .....	38
5.3.4	Importing Users .....	39
5.3.4.1	Having Subversion Users Change Their Passwords .....	39
6	About Access Control Lists .....	40
6.1	How WANdisco Enforces Rules .....	40
6.2	Toggling the ACL Display .....	41
6.3	Creating ACLs .....	43
6.4	Toggling the Use of Access Control Lists .....	43

# Contents, cont'd

7	About Audit Reports	44
7.1	Setting Up the Reports	44
7.1.1	Using the Import Tool	45
7.2	Configuring Audit Properties	46
7.3	Running a Report	48
7.3.1	Report Types	49
7.3.1.1	The User Report	49
7.3.1.2	The Transaction History Report	50
7.3.1.3	The Access Violation Report	50
7.3.1.4	The File Report	50
8	Procedures	51
8.1	Preventing Subversion Users From Making Transactions	51
8.2	Changing the prefs.xml File	51
8.3	Verifying That Access Control is Working	52
8.4	Installing a .jar File Patch	53
8.5	Setting WANdisco to Start Up on System Boot	54
8.6	Setting WANdisco Up as a Windows Service	55
8.7	Changing SVN Port on Unix Flavor	56
9	Troubleshooting	57
9.1	How Do I Get WANdisco Support?	57
9.2	Apache and SVNKit	57
9.2.1	SVNKit and Connection Pooling	57
9.2.2	Tuning Values to Optimize Your Configuration	57
9.3	Error Messages	58
9.3.1	Missing License Key File	58
9.3.2	Client could not read status line: Server Closed Connection	59
10	Frequently Asked Questions	60
10.1	Why Are So Many Java Processes Running?	60
10.2	Can I Store Logs or Content on NFS?	60
10.3	Why is Set Up Configuring IP Addresses as 0.0.0.0?	60
10.4	WANdisco Authentication	61
10.5	Apache 2.2 with SVNDAV on Windows	61

## Contents, cont'd

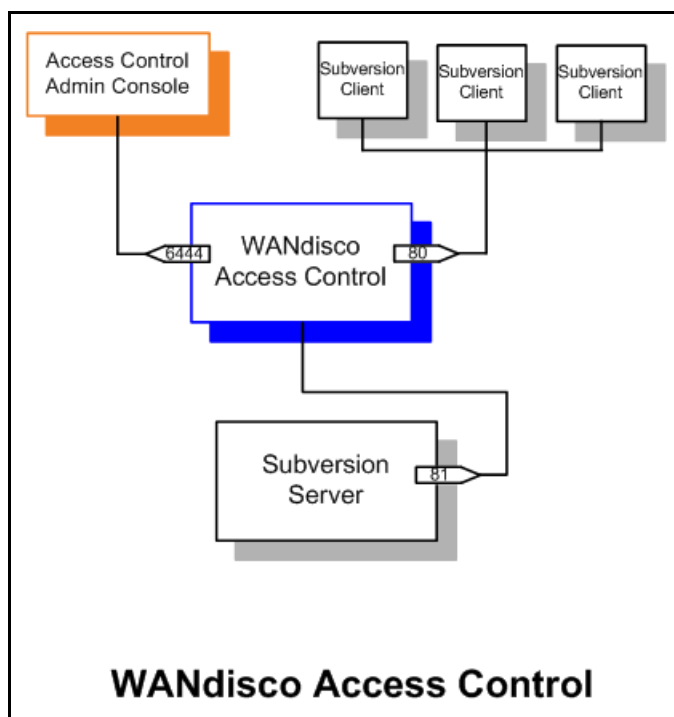
10.6 Setting Up Apache for SVN-DAV .....	62
10.7 Encryption Around WANdisco Protocol .....	64
10.8 How Do I Restrict Direct Access to My Repository? .....	64
10.9 About WANdisco Log Files .....	65
<b>Appendix A - Installing Java and Perl .....</b>	<b>A - 1</b>

# 1 Introduction

---

Welcome to the WANdisco world of security. WANdisco Access Control for Subversion allows Subversion users to transparently connect to a Subversion repository through a network proxy acting as a security agent. Subversion users connect to WANdisco on the standard Subversion port 80. WANdisco Access Control in turn connects to the underlying Subversion server through port 81. Subversion users never directly access the repository. WANdisco Access Control enables you to implement comprehensive security features, including:

- Setting up Access Control Lists (ACLs) across multiple SVNROOTs and modules
- Role-based ACLs
- Hierarchical Groups and Privileges
- Roles for separate actions:
  - ◆ List
  - ◆ Read
  - ◆ Write
  - ◆ Tag
  - ◆ Delete
  - ◆ Admin (all actions)
- Access control of files, directories, and modules
- Ability to limit access via network masks or IP address
- Full Perl-style regular expression support
- Two default reports: all users, and all groups, and the resources they can access



WANdisco provides the Admin Console, a web-based user interface, to administer and monitor the replication group.

A stand-alone Subversion instance is hosted on a J2EE-compliant application server. Users access the application with a standard web browser and Subversion or HTTP protocol (via the HTTP port, by default 80). WANdisco Access Control configures an additional port (by default 6444) for the Admin Console communications (known as DConE) using the DConENet protocol.

## 1.1 WANdisco Access Control Concepts and Terms

Here are definitions for commonly used terms in WANdisco Access Control:

Login id	The actual Subversion account name that can be successfully authenticated by Subversion or SSH daemon (if using the ext SSH protocol). The Subversion login id is also the primary key for a user in the Access Control user database.
Principal	Principal can be any valid user or group. After authentication, Access Control maps a login id to a set of rules that include the actual user and all its associated groups and sub-groups.
Resource	Resource is a file, directory, module or the SVNROOT itself. Resource patterns can be specified as Perl-style regular expressions in the ACL. All directory paths should be specified in the slash-terminated form. For example, specify <code>/a/b/c/</code> , not <code>/a/b/c</code> .
IP Mask	A Perl-style regular expression specifying the Subversion client's IP address. It is used in the ACL to restrict access to a specific client network, subnet or a machine.
Privilege	Privileges are needed by a user to execute specific Subversion commands. WANdisco Access Control supports these privileges: <ul style="list-style-type: none"><li>■List</li><li>■Read</li><li>■Write</li><li>■Copy</li><li>■Delete</li><li>■Admin</li></ul>

Admin serves as a privilege, a role, and a group. It is meant for a System Administrator who has no restrictions. Do not create ACLs for an Admin user.

### 1.1.1 Users, Roles, and Groups

You set up Subversion users by assigning them roles. Each role defines specific privileges. The fundamental permissions for a user are defined by their role. Then, you assign each user to a group, and give that group access to the desired files and directories (referred to as "resources").



The group level defines a user's access to the resources. With this structure, you can have multiple projects with complex resource definitions, allowing users to perform tasks associated with their roles' permissions on those resources.

For example, you create two users: Mary, who has the developer role, and Bob, who has the QA role. Next, you create a group named *project1* and specify which Subversion directories the group has access to (you define the resources). So Mary is allowed to perform write operations and Bob is allowed to perform read and copy operations on those directories.

**NOTE:**

---

Any user you add to Access Control must already exist in the authentication database used by the Subversion server. WANdisco offers an import tool to do some external SQL verifying of user lists, but it is up to the Access Control administrator to control the validity of the users.

---

### 1.1.2 Access Control Lists

For some customers, roles and groups provide enough control. However, you can further refine the granularity of control for the roles, groups and resources by creating Access Control Lists (ACLs), also called rules.

For example, in the group *project1*, if you don't want Mary to have access to one sub-directory in the resources you have defined, you must create an ACL to prohibit her write privilege to that sub-directory.

You can create as many ACLs as you need, however you should have a comprehensive plan to avoid creating conflicting ACLs.

### 1.1.3 Reports

Access Control offers two reports, Users and Groups. You may want to perform more in depth audits of the ACLs. This requires importing Access Control information into an SQL database.

WANdisco offers an import tool that automatically creates the table schema in that database. The import tool uses standard SQL syntax, and makes use of the system function `FROM_UNIXTIME`. Please ensure your database version supports it. MySQL and Microsoft SQLServer both support the `FROM_UNIXTIME` function.

## 1.1.4 Sequence For Implementing Access Control

The general sequence for implementing Access Control is listed below. It is a fairly simple process. Setting up any ACLs can be the most complicated.

- Step 1     define roles
- Step 2     create groups
- Step 3     add users
- Step 4     set up ACLs, if desired

## 1.1.5 Perl Style Regular Expressions and Syntax

You can use Perl style regular expressions wherever patterns are allowed throughout the Admin Console. Principal (user/group) or IP patterns, for instance - `engineering.*` (note the dot) or `217.[0-9]+` are all valid patterns.

With the Perl regular expression syntax, if you need to use the '.' (dot) character literally, you need to escape it with a backslash, otherwise '.' (dot) matches any character. To learn more about regular expressions, read [this tutorial](#).

For example, to allow any user with an IP address that begins with 192.168, enter the value: `192\.168.*`. Note that the period character must be prefixed by a backslash. With regular expressions, the period character is used to represent a character that can have any value.

## 1.2 Terms

You should familiarize yourself with these terms.

TERM	DEFINITION
GUID	Globally Unique Identifier. WANdisco Subversion Access Control assigns a GUID to the site where WANdisco is on installation.
prefs.xml	The preferences files contain information on the Access Control. The preference file is located in <code>svn-security/config</code> .
SCM Repository	Software Configuration Management repository like Subversion
Subversion Server	A network server that provides remote access to a Subversion Repository
installDir	this is the installation directory for WANdisco.

## 2 Recommended Deployment Practices

---

Please read and follow these requisites to ensure a successful installation and use of Access Control.

### 2.1 Access Control Administrator Pre-requisites

This guide is intended for an Subversion administrator or a user who is reasonably comfortable with:

- Setting up an Subversion based repository
- Configuring inetd/xinetd service on Unix/Cygwin or Windows service
- Installing Perl and required Perl modules
- Installing Java
- Installing Apache server if using HTTP protocol
- Unix or Windows system administration
- familiarity with regular expressions

If you don't meet the above pre-requisites, you may want to contact your Subversion administrator or request that WANdisco perform a professional install for you.

### 2.2 Physical Environment

WANdisco strongly recommends that you follow these guidelines:

- Subversion and Access Control running on the same server (so that WANdisco can control user password files)
- a running pre-configured server with
  - ◆ a command line zip/unzip utility
  - ◆ Java (see [Appendix A - Installing Java and Perl](#))
  - ◆ Perl (see [Appendix A - Installing Java and Perl](#))
  - ◆ browser with network access
- e-mail from WANdisco containing the WANdisco file link and attached production licence key file

## 2.3 WANdisco and Subversion Password Files

WANdisco offers to control the Subversion password files, thereby eliminating duplicate user entries for either program. If WANdisco controls the password files, any user you set up in WANdisco is also set up in Subversion. The majority of WANdisco customers elect to have WANdisco handle the passwords for both. The Admin Console offers an easy way to manage users and passwords.

During installation, you identify the Subversion password file's location, and WANdisco incorporates it into Access Control. You can also choose to bulk import your Subversion users into WANdisco, however this resets user passwords to their email addresses.

If you do elect to have WANdisco control the Subversion password files, all users must be entered into WANdisco. Note that for DAV, WANdisco does not handle user authentication, it just acts as a proxy between users and Subversion.

## 2.4 Firewalls and Virus Scanners

### 2.4.1 Firewalls

You must determine if WANdisco sits inside a firewall or outside of one. If WANdisco sits inside a firewall, it is untouched by the firewall and you need take no action.

However, if WANdisco sits outside a firewall, you must configure the firewall so that the port numbers you specify during installation are not blocked or filtered.

### 2.4.2 Virus Scanners

If you have a virus scanner running on your network, you must configure it to not filter traffic on the ports you specify during installation.

## 2.5 Deployment Checklist

You may be familiar with this checklist from an evaluation copy of Subversion Access Control. It is included here as reference.

<b>System Setup ❖ All sites must share the same operating system</b>	
Supported Operating Systems	<p>Fedora (32 or 64 bit): 6, 7, 8, 9            Red Hat Linux Enterprise Server (32 or 64 bit): 4, 5.2            Sun Solaris (32 or 64 bit): 9, 10            Linux: Linux kernel 2.6 or higher            CentOS-4            Windows Server, (32 or 64 bit) 2003</p> <p>Please <a href="#">read this</a> for more information on NPTL.</p> <p><b>Note:</b> VMware has a tendency to become unresponsive due to memory paging issues even without WANdisco present. Extra tuning may be needed to ensure optimal performance.</p>
Subversion Server Version	1.3 and above. If you are using Subversion 1.5.4, use version 1.3.0 of Apache Portable Runtime.
Subversion Client Version	compatible with local Subversion servers
System Memory	<p>Ensure RAM and swapping containers are at least three or four times the largest Subversion file you have.</p> <p>Recommended: 1 GB RAM; 2 GB swapping container</p>
Disk Space	<ul style="list-style-type: none"> <li>• CVS: depends on the number of projects and issues</li> <li>• Access Control Transaction Journal: Recommended - equivalent of seven days of changes</li> </ul>
File Descriptor limit	Ensure hard and soft limits are set to 64000 or higher. Check with the <code>ulimit</code> or <code>limit</code> command.
Journaling File System	Replicator logs should be on a journaling file system, for example, ext3 on Linux or VXFS from Veritas. Notes: NTFS is not a journaling file system: ext4 is a journaling file system, however WANdisco does not support its use because of its deferred writes.
Maximum User Process Limit	At least three times the number of Subversion users.
Java	Install JDK 1.5.0. <b>Note:</b> There should not be any spaces or control characters in the path where Java is installed. For example, <code>c:\Program Files\java</code> does not work with WANdisco as a JAVA install directory. See <a href="#">Appendix A - Installing Java and Perl</a> .
Perl	<p>Install version 5.6.1 or later. See <a href="#">Appendix A - Installing Java and Perl</a>.</p> <p>Perl::DBI module for Audit Reports other than Users and Groups</p>

<b>Network Setup</b>	
Reserved Ports (i.e. 6444, another for synchronizing)	Subversion Access Control needs a dedicated port for DConENet (replication protocol) as well as HTTP protocol (for the Admin Console). WANdisco also recommends having a port available in case you have to copy (rsync) the repository from one site to another. If your network has a firewall, notify the firewall of the port numbers.
Firewall and virus scanner	Notify the firewall and any virus scanners of the Subversion Access Control port numbers.
Persistent Connection Keep Alive	Ensure VPN doesn't reset persistent connections for WANdisco, or else ensure there are no RST bugs
DNS Setup	Best to use IP address for WANdisco related hostnames, or else ensure DNS availability
<b>WANdisco Setup</b>	
Agreement Threads	Tune based on number of concurrent Subversion writers
Reader/Writer Network IO Thread Pool	Tune based on Subversion client connection rate, file transfer rate
ConnectionKeepAlive timeout	Tune inactivity timeout for persistent DConENet/DFTP connections based on VPN/WAN router set up
Message Queue Max Thread Pool Size	Tune based on Subversion write concurrency
Maximum connections per IO thread	Tune if active Subversion user population is large (greater than 100)
Disk space for recovery journal	Provision large disk for <code>logs/tmp</code> , at least number of commits within a two to four hour window
Reader/Writer Network IO Thread Pool	Tune based on client connection rate, file transfer rate
ConnectionKeepAlive timeout	Tune inactivity timeout for persistent DConENet/DFTP connections based on network delays
Admin Email Address	To generate email notifications from Access Control. Requires <code>/usr/sbin/sendmail</code> .
Audit Reports	For reports other than Users and Groups, you'll need to use a database such as MySQL, and php.
Notify all users that they must flush their client cache.	
<b>Apache 2 Setup (for http:// access)</b>	
Apache version	All sites have the same version, 2.2.3 and above.
Apache modules version	All sites have the same version of <code>mod_dav</code> and <code>mod_svn_dav</code>  For audit reports other than Users and Groups, use php module (usually part of php installation).

Apache KeepAlive	Ensure proper keep-alive settings. See <a href="http://httpd.apache.org/docs/2.2/mod/core.html">http://httpd.apache.org/docs/2.2/mod/core.html</a> .
Require valid user for write methods	Ensure that all WebDAV methods require authentication for SVN-DAV protocol
Using port 80 for WANdisco	Std Port 80 avoids confusion, change default Apache port if using 80
Apache server port	Non-standard Apache server port to avoid conflict with replicator port?
File Permissions in svnroot	See this article ( <a href="http://www.reallylongword.org/articles/svn/">http://www.reallylongword.org/articles/svn/</a> )

## 2.6 New Subversion Repository

If you are creating a new Subversion repository, please follow the Subversion documentation at <http://svnbook.red-bean.com>.

## 2.7 Using HTTP with Apache

Apache's configuration needs to be tweaked to work in a performant manner with Subversion (this is independent of Subversion Access Control). Perform the following steps.

- Step 1 Change Apache's connection keep-alive settings to allow long lived HTTP connections. Add this to the Apache configuration file `conf/httpd.conf` or included `conf/extra/httpd-defaults.conf`. For instance,

```
$ vi conf/httpd.conf
...
# Various default settings
Include conf/extra/httpd-default.conf
...
$ vi conf/extra/httpd-default.conf
...
#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300000
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection).
#
KeepAlive On
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
#
MaxKeepAliveRequests 0
#
```



```
# KeepAliveTimeout: Number of seconds to wait for the next request from
the
# same client on the same connection.
#
KeepAliveTimeout 500000
...
```

- Step 2** Ensure the SVN DAV settings in Apache's configuration files are exactly the same at all sites. The top level location URI prefix should be the same. We recommend copying the current `conf` file and, then changing the `host:port` settings. For instance, here is a `conf` file snippet with Apache virtual hosts (Note: you do not have to use Apache virtual hosts, this is only an illustration):

```
# Site A
$ cat conf/extra/httpd-svn-dav.conf
...
NameVirtualHost site-a:8181
<VirtualHost site-a:8181>
<Location /dir0>
DAV svn
SVNPath /home/site-a/svnroot
AuthType Basic
AuthName wandisco
AuthUserFile /home/site-a/apache2/dist/conf/htpasswd
Require valid-user
</Location>
</VirtualHost>
...
# Site B
$ cat conf/extra/httpd-svn-dav.conf
...
NameVirtualHost site-b:9191
<VirtualHost site-b:9191>
<Location /dir0>
DAV svn
SVNPath /home/site-b/svnroot
AuthType Basic
AuthName wandisco
AuthUserFile /home/site-b/apache2/dist/conf/htpasswd
Require valid-user
</Location>
</VirtualHost>
...
```

- Step 3** The Apache user-names and passwords should match at all sites. Subversion Access Control's license manager requires a valid username inside the HTTP authorization header to be passed for all DAV commands. This typically means ensuring a `Require valid-user` line is specified in the Apache SVN DAV configuration section.

- Step 4      Subversion Access Control requires all HTTP methods to be authenticated.

## 3 Installation

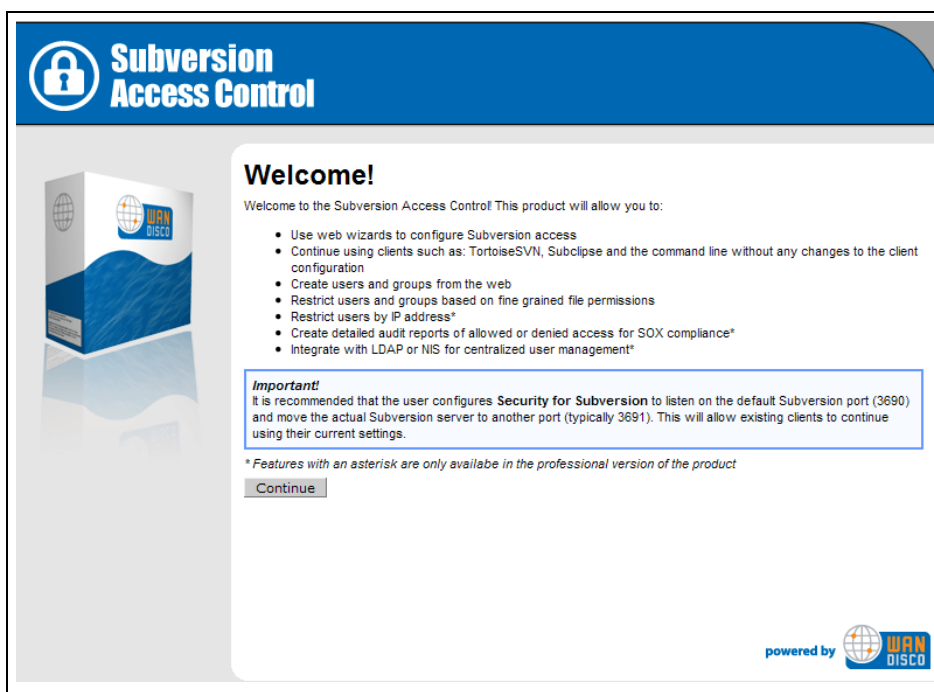
### 3.1 First Time Installation

You must run the installation program at the Access Control server. Ensure that Subversion is running.

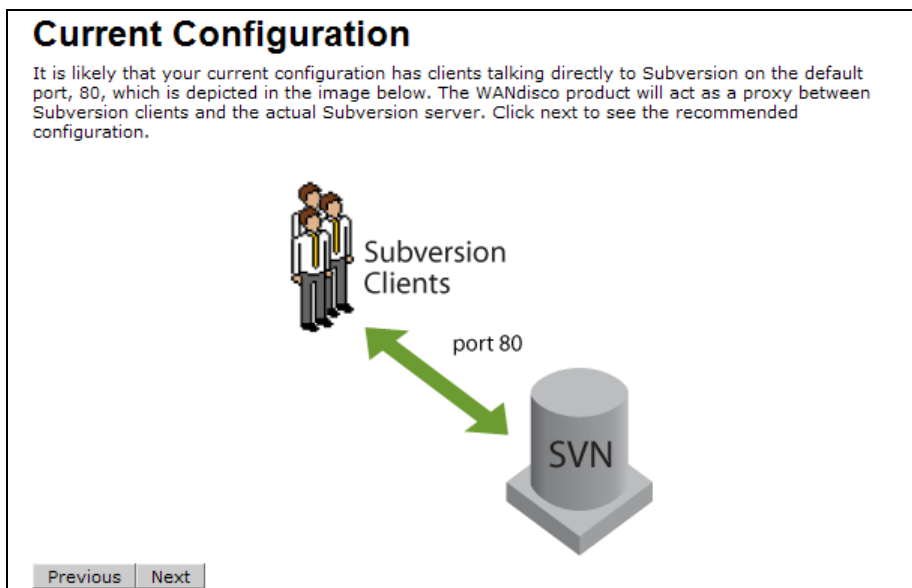
- Step 1 Save the `svnsec.tar.gz` file.
- Step 2 Unzip or untar the file, which produces a folder, `svn-security`.
- Step 3 Copy the licence evaluation key file to the `config` folder in the `svn-security` folder.
- Step 4 At the command prompt (or editor), go to `svn-security/bin`.
- Step 5 Type
- ```
perl setup
```

The last line of text returned contains a WANdisco URL.

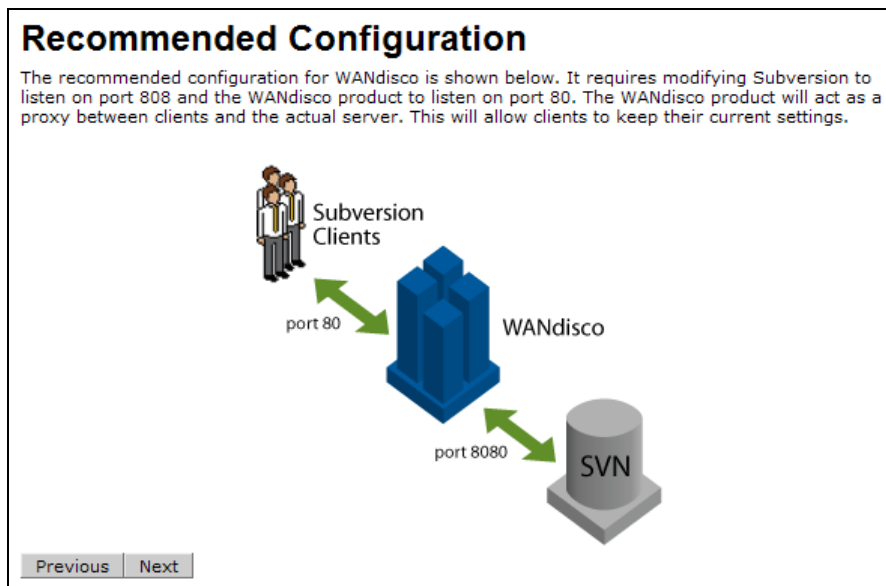
- Step 6 Copy the URL returned in the last step and paste it into a browser. The WANdisco Welcome page appears.



- Step 7 Click **Continue** at the Welcome text.
- Step 8 Read the User Licence Agreement. You must agree to the terms to continue.
- Step 9 The Current Configuration page appears. After reading it, click **Next**.

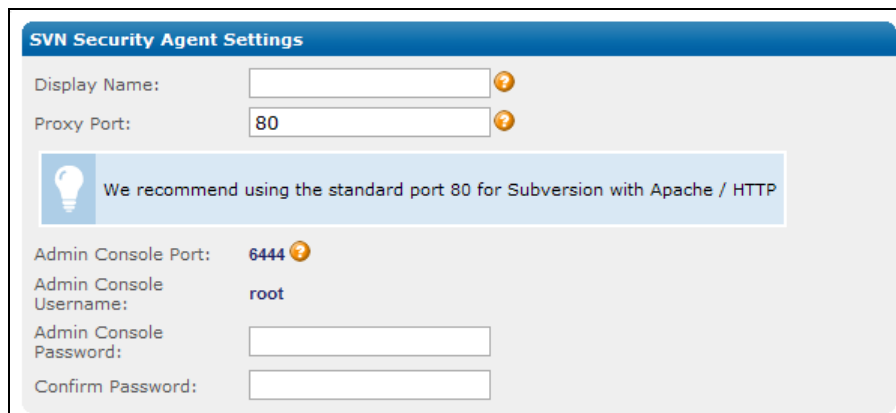


- Step 10 The Recommended Configuration page appears.



- Step 11 The illustration shows how WANdisco becomes the default Subversion port, and you will assign a different port for the Subversion server. Click **Next**.

### 3.1.1 Security Agent Settings



Step 12 In the Display Name field, enter a name for the Access Control proxy.

Step 13 In the Proxy Port field, enter a port number. WANdisco recommends using 80 as the port number if you are using Apache/HTTP.

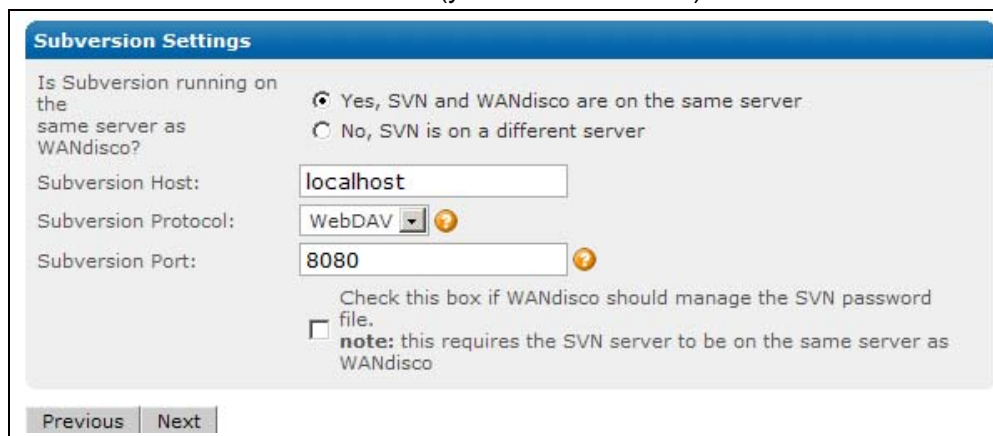
Step 14 In the Admin Console port field, the port number 6444 is reserved for WANdisco.

Step 15 The Admin Console username is reserved as `root`.

Step 16 Enter and confirm a root password.

### 3.1.2 Subversion Settings

Step 17 Check **yes** or **no** to tell WANdisco if Subversion is running on the same server as WANdisco (your current server).



- Step 18 In the Subversion Host field, enter a name. If you answered `yes`, WANdisco and Subversion are on the same server, you can use `localhost` as the name.
- If you answered no, you must provide the IP of the Subversion host.
- Step 19 Select the Subversion protocol you are using.
- Step 20 In the Subversion Port number, enter in a different number than you specified in Step 13. For WebDAV, WANdisco recommends 8080.
- Step 21 Specify if WANdisco is going to manage the Subversion password file. (Subversion and WANdisco must be on the same server for WANdisco to manage the file.)
- Step 22 If you have SVN and WANdisco on the same server, and you want WANdisco to manage the SVN password file, browse to the location of the SVN password file.
- Step 23 Click **Next**.

### 3.1.3 Checklist

#### Checklist

This checklist is designed to provide step-by-step instructions for installing WANdisco as quickly and easily as possible. Click on the checkbox beside each task to indicate that it has been completed. Links are provided for the more complex tasks. If further assistance is needed contact [WANdisco support](#).

**System Settings**

- Verify that JDK 1.5\_03 or above is installed on the server(s) where WANdisco will be installed
- [Verify System Setup](#)
- [Verify Network Setup](#)

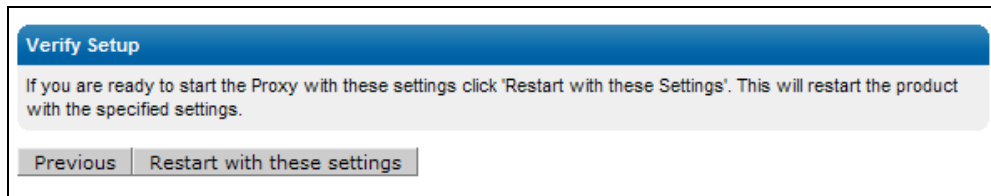
**Subversion with Apache Settings**

- Ensure apache is running on port 8080
- Ensure port 80 is available for the WANdisco proxy
- [Review the article: Setting up Apache for SVN DAV](#)
- [Check file permissions in the SVNROOT directory](#)

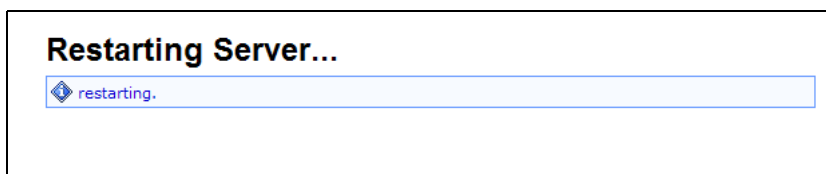
Previous Next

Step 24 Read over each item, and check each checkbox. You must be logged in to the WANdisco support site for the links to work.

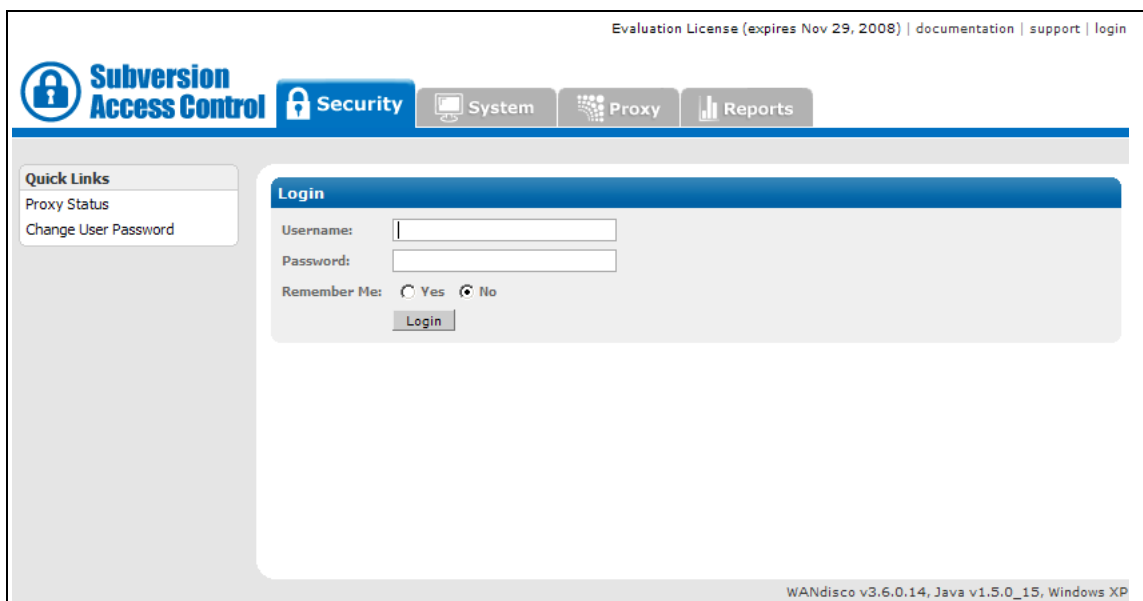
Step 25 Click **Next**. The Verify Setup page appears.



Step 26 Click **Restart with these settings**. The installer configures and restarts your machine.



The server restarts with Access Control up and running.



Step 27 Log in with the username and password you specified in step 16.

You are now ready to set up Access Control with users, roles and groups.

The installer creates a directory called `svn-security` that has these directories inside.

| DIRECTORY | CONTENTS                                                                                                                                                                 |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| audit     | Contains                                                                                                                                                                 |
| bin       | Contains scripts like <code>svnsecurityagent</code> , <code>shutdown</code>                                                                                              |
| config    | Contains the <code>prefs.xml</code> file used to configure Access Control.                                                                                               |
| lib       | Contains the <code>jar</code> files and DLLs that are required to run the product.                                                                                       |
| docs      | Contains this <i>Administration Guide</i> in PDF format.                                                                                                                 |
| logs      | Contains the <code>pid</code> file, log files and other temporary files. Access Control's log file is named <code>SVNProxyServer-prefs.log.0</code> .                    |
| systemdb  | Contains the system database with its transaction journal. <b>Warning:</b> Deleting or modifying files from <code>systemdb</code> will likely corrupt your installation. |

## 3.2 Installing Upgrades

### NOTES:

---

This procedure involves taking Subversion offline. Please follow your company procedures about notifying Subversion users of down time.

---

### 3.2.1 Disconnect SVN Users

Step 1 After notifying Subversion users of the downtime, stop Access Control. On the Proxy page, select **Stop Proxy**.

### 3.2.2 Export WANdisco Data

Step 2 On the System page, select **Export Settings**. This is the ACLs, and user and group information.

### 3.2.3 Stop WANdisco

Step 3 Stop WANdisco Access Control. On the Proxy page, select **Shut Down Node**.



### 3.2.4 Preparing the Install Node

Step 4 On the original install node, zip these directories:

```
config
```

Step 5 Delete this directory:

```
systemdb
```

Step 6 In the `svn-security/config` directory, delete these directories:

```
membership  
security  
passwd
```

### 3.2.5 WANdisco-supplied .jar and License Key Files

Step 7 Save the `svnsec.tar.gz` file.

Step 8 Verify the md5 checksum. For Unix, type

```
md5sum
```

Step 9 Copy the jar file to the `svn-security/lib` directory.

Step 10 Unzip or untar the file.

Step 11 Copy the licence evaluation key file to the **config** folder in the `svn-security` folder.

### 3.2.6 Running the Install Program

Step 12 Run this command on the original install site. The install program automatically populates all previous configuration information.

```
svn-security/bin/setup
```

The Setup page appears.

Step 13 Click **Next** to continue.

Step 14 Make changes as needed. The Admin Console appears.

### 3.2.7 Importing WANdisco Data

- Step 15 Import WANdisco data. On the System page, select **Import Settings**. Import the file you exported in step 2.

### 3.2.8 Verifying Installation

- Step 16 Ensure Access Control is active. Go to the Proxy page and verify that the value for the `listening` field is `yes`.

You have successfully installed an Access Control upgrade.

## 4 Using the Admin Console

The Admin Console is a simple interface that allows you to configure Access Control.

This chapter describes how to use the two Access Control pages and lists the commands on each. For information on how to set up users, roles, groups and ACLs, see the next two chapters, [About Users, Roles and Groups](#) and [About Access Control Lists](#).

### 4.1 Starting the Admin Console

To start the Admin Console, in a browser's address bar, type

*http://<IP address>:<WANdisco port number>*

The Admin Console's Home page appears.

Access Control's Admin Console has four pages, identified by their tabs: the Security, System, Proxy and Reports pages.



## 4.2 The Security Page

### NOTE:

---

Password fields appear for users only if you chose to have WANdisco control the Subversion password file during installation. For DAV, WANdisco does not handle the user authentication.

---

### Role Administration

|             |                                                                                                                                                                                                                                               |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Create Role | Create new roles and assign them privileges. Subversion permissions are: list, read, prewrite, write, delete, copy, admin. For a complete discussion of roles and permissions, see <a href="#">Chapter 5, About Users, Roles and Groups</a> . |
| List Roles  | Display all roles: default and any you create. Privileges are also displayed. To delete a role, check any role's checkbox and click <b>Delete Selected</b> .                                                                                  |

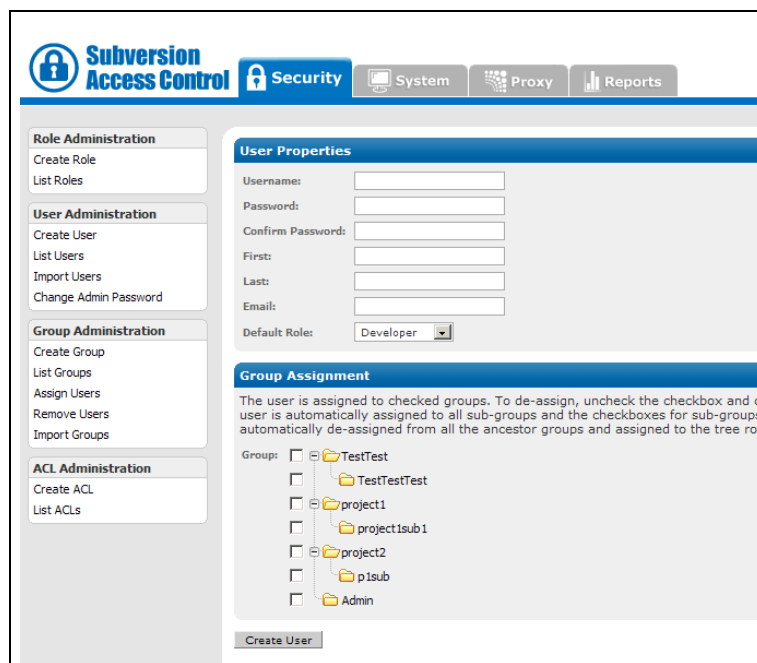
### User Administration

|                       |                                                                                                                                                                                                   |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Create User           | Create any Subversion user.                                                                                                                                                                       |
| Username              | Enter in the Subversion user's username.                                                                                                                                                          |
| Password              | Enter the user's password.                                                                                                                                                                        |
| Confirm Password      | Confirm the password.                                                                                                                                                                             |
| First Name            | Enter the user's first name.                                                                                                                                                                      |
| Last Name             | Enter the user's last name.                                                                                                                                                                       |
| Email                 | Enter the users email address.                                                                                                                                                                    |
| Default Role          | Give the user a default role.                                                                                                                                                                     |
| Group Assignment      | If desired, assign the user to an existing group.                                                                                                                                                 |
| List Users            | This command displays all users.                                                                                                                                                                  |
| Import Users          | You can import an existing list of users. The import file must be a comma delimited text file, of the format <code>username, last-name, firstname, email[, group1[, group2, ... groupN]]</code> . |
| Change Admin Password | You can change the WANdisco Admin password with this command.                                                                                                                                     |

### Group Administration

|                    |                                                                                                                                                         |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| Create Group       |                                                                                                                                                         |
| Name               | Required field. You must name the group.                                                                                                                |
| Description        | Enter in a description for the group.                                                                                                                   |
| Client IP Pattern  | Optional. You can allow certain IP patterns. Use regular expressions. If you do use this option, no other client IPs are allowed without specific ACLs. |
| Rule               |                                                                                                                                                         |
| File / Dir Pattern | Browse to the file or directory pattern for this group, and specify either Allow or Deny. Use regular expressions.                                      |
| add allow          | identify any files and directories that this group has access to                                                                                        |
| add deny           | identify any files and directories that this group does not have access to                                                                              |

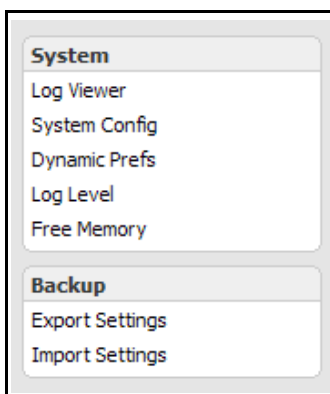
|                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>Create Group</b></p> <p>List Group</p> <p>Assign Users</p> <p>Remove Users</p> <p>Import Groups</p>                                                    | <p>Select this command once you have defined a group.</p> <p>This command shows all the groups. You can list all users in each group.</p> <p>This command allows you to assign users to groups. If a user is already in a group, his or her name does not appear in the list of available users.</p> <p>Use this command to remove users from a group.</p> <p>You can import a list of existing groups. The import file must be a comma delimited text file, of the format <code>groupname,parent-name[,description]</code>. If there is no parent name, specify <code>null</code>.</p>                                                                                                                                                                                                                                                                                                                                                             |
| <p><b>ACL Administration</b></p>                                                                                                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <p>Create ACL</p> <p>User / Group Rule</p> <p>Privilege Operate On</p> <p>IP Pattern</p> <p>File / Dir Pattern</p> <p><b>Create ACL</b></p> <p>List ACLs</p> | <p>This menu item can be toggled. See <a href="#">6.2, Toggling the ACL Display</a>.</p> <p>Create an ACL with this command. Create more than one at a time, use List ACLs. For a complete discussion on ACLs, see Chapter 6, <a href="#">About Access Control Lists</a>.</p> <p>Specify the user or group for this ACL.</p> <p>Specify <code>allow</code> or <code>deny</code>.</p> <p>Identify a privilege for this ACL.</p> <p>Specify if this ACL operates on a single user or the group.</p> <p>Optional. You can identify an IP pattern for this ACL. If you do identify an IP pattern, all other IPs are excluded, unless you create an allow ACL for a specific IP address.</p> <p>Identify the file or directory pattern for this ACL.</p> <p>Select this command to create the ACL.</p> <p>This command lists all existing ACLs. You can create, edit or delete ACLs with this command. Use this command when creating multiple ACLs.</p> |

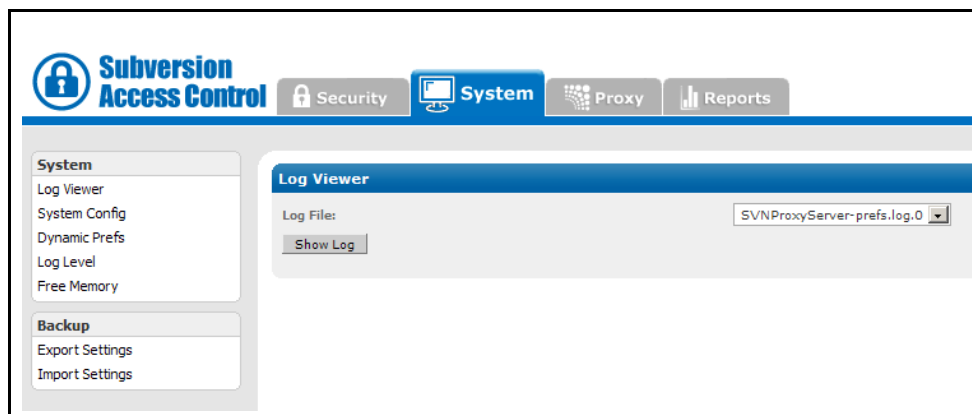


## 4.3 The System Page

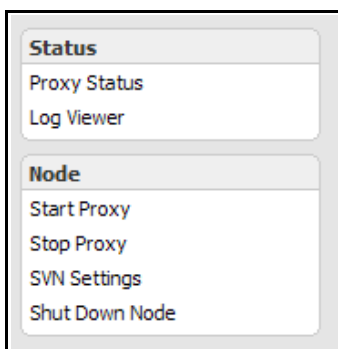
### System

|                          |                                                                                                                                                                                                                        |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Log Viewer               | View the logs.                                                                                                                                                                                                         |
| groups-reports.txt       | A simple report of created groups and their resources.                                                                                                                                                                 |
| SVNProxyServer-prefs.log | The WANdisco log file.                                                                                                                                                                                                 |
| users-reports.txt        | A simple report of users and their groups and resources.                                                                                                                                                               |
| web proxy.log            | A log from the installation.                                                                                                                                                                                           |
| <b>Show Log</b>          | Click this to display the log.                                                                                                                                                                                         |
| System Config            |                                                                                                                                                                                                                        |
| Show ACLs?               | Check <b>Yes</b> or <b>No</b> to show ACLs. ACLs appear on the Group Properties page in the Security tab. This also causes the ACL Administration menu to display on the Security tab.                                 |
| Display Sibling Groups?  | Check <b>Yes</b> or <b>No</b> to display sibling groups. <b>No</b> is recommended.                                                                                                                                     |
| Dynamic Prefs            | This is used internally for troubleshooting.                                                                                                                                                                           |
| Log Level                | WANdisco uses one log, and the default level is <b>info</b> . The levels vary from <b>severe</b> , where you get only the most severe warnings, to <b>finest</b> , which logs every action.                            |
| Free Memory              | This command frees the memory (GB stands for garbage collection) for the current node. The command occurs when you click on this menu selection. The display shows information on the command that was just performed. |
| max mem used by JVM      | the maximum memory that JVM can use on the current node                                                                                                                                                                |
| free memory before GC    | the amount of free memory before you ran this command                                                                                                                                                                  |
| free memory after GC     | the amount of free memory after you ran this command                                                                                                                                                                   |
| memory freed             | the total amount of memory freed at the command's completion                                                                                                                                                           |
| Backup                   |                                                                                                                                                                                                                        |
| Import Settings          | This command allows you to import WANdisco settings, including all users, roles, groups and ACLs.                                                                                                                      |
| Export Settings          | This command allows you to export WANdisco settings, including all users, roles, groups and ACLs, for later importation into a WANdisco product.                                                                       |





## 4.4 The Proxy Page



### Status

#### Proxy Status

Node Name

This displays the node name.

SVN Client Port

This displays the SVN Client Port.

SVN Server

This displays the SVN Server name.

listening

This signifies whether WANdisco is listening to Subversion client transactions.

WANdisco Install

This is the WANdisco installation directory.

Web DAV Version

This gives the web DAV version specified on installation.

SVN Password File

This gives the path to the password file specified on installation.

GUID

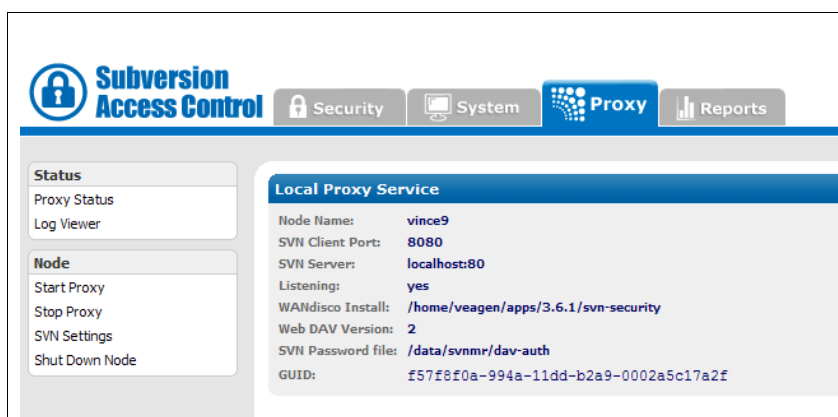
This is WANdisco's identification for the server WANdisco runs on.

#### Log Viewer

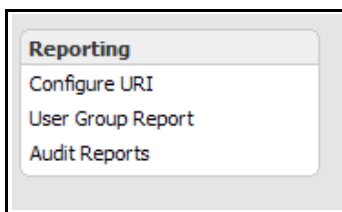
Provides access to the same logs discussed in [Log Viewer](#) on the System page.

**Node**

- Start Proxy                      This command starts Access Control.
- Stop Proxy                      This command stops Access Control.
- SVN Settings                    These fields are blank until WANdisco needs to know further information about your Subversion implementation. A WANdisco message informs you when to fill in the fields.
- Shut Down Node                This command shuts down the Access Control server. To restart Access Control, at the command line, go to `svn-security/bin/` and type `svnsecurityagent`.



## 4.5 The Reports Page



**Reporting**

- Configure URI                    Refer to [Chapter 7, About Audit Reports](#).
- User Group Report              Generate these reports, and view them with Log Viewer in the System and Proxy tabs.
- Audit Reports                    Refer to [Chapter 7, About Audit Reports](#).



## 5 About Users, Roles and Groups

---

This chapter provides information on setting up users, roles and groups for Access Control. Most customers find that managing users' roles and groups offer enough access control. However, you can further refine Access Control with specific Access Control Lists, discussed in Chapter 6, [About Access Control Lists](#).

Access Control initially does not allow any user access to any resource. By default, all users are denied. This is essential for security: it closes the window of vulnerability that would allow everyone full access between the time WANdisco is first installed and the time it takes an administrator to create access rules. In order to grant access, the administrator has to explicitly create roles, groups (which define resources) and users.

You should be familiar with the Admin Console, described in Chapter 4, [Using the Admin Console](#).

Resources assigned to subgroups are given to users who are members of parent groups because the users are automatically members of the subgroup. So, membership goes down the tree and inheritance, as a result, goes up.

### 5.1 Managing Roles and Permissions

Access Control's roles are based on Subversion permissions. The default roles are:

- ◆ list
- ◆ read
- ◆ prewrite
- ◆ write
- ◆ delete
- ◆ copy
- ◆ admin

The following table offers a mapping of actual Subversion commands to the minimum permission needed to execute them.

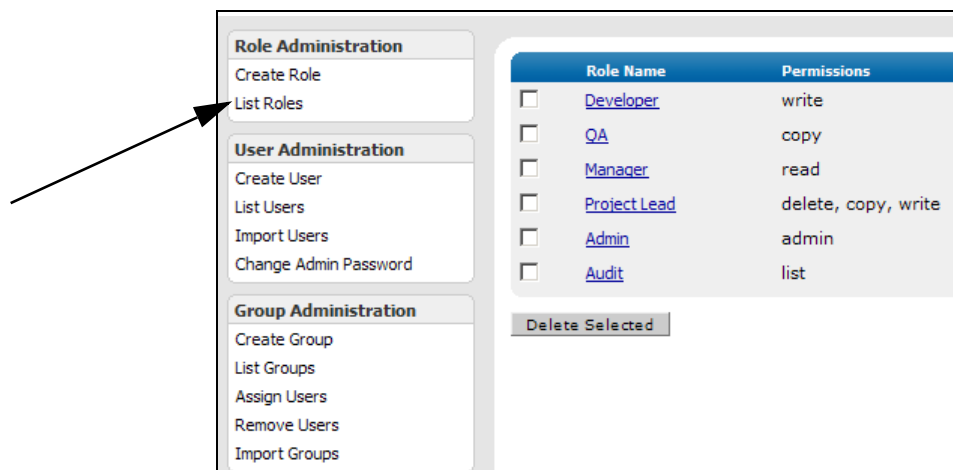
| Subversion Command | Permission Required |  | Subversion Command | Permission Required |
|--------------------|---------------------|--|--------------------|---------------------|
| info               | List                |  | revert             | Read                |
| log                | List                |  | annotate           | Read                |
| ls                 | List                |  | propget            | Read                |
| status             | Read                |  | proplist           | Read                |
| cat                | Read                |  | commit             | Write               |
| diff               | Read                |  | import             | Write               |
| checkout           | Read                |  | add                | Write               |
| cleanup            | Read                |  | lock               | Write               |
| update             | Read                |  | unlock             | Write               |
| revert             | Read                |  | copy               | copy                |

Access Control comes with a few default roles with existing permissions. You can modify these roles as you wish. You can also create new roles. The permissions are inherited, meaning if a role has the write privilege, it also has the list and read permissions as well.

The roles work with groups, which you defined as files or directories. So the roles are applied within the groups (the defined files or directories).

| Default Role | Privileges - Inherited Hierarchy |
|--------------|----------------------------------|
| Audit        | list                             |
| Manager      | list, read                       |
| Developer    | list, read, write                |
| QA           | list, read, copy                 |
| Project Lead | list, read, delete, copy, write  |
| Admin        | list, read, delete, copy, write  |

The **List Roles** command, under Role Administration, shows all roles: the default roles and any you have created. The permissions for the roles are also listed.



**NOTE:**

---

Admin serves as a permission, a role and a group. The Admin privilege has no constraints on it whatsoever. An admin has full permission to everything in the repository. It is designed to be used for a System Administrator.

If you assign a user the Admin role, or give a user Admin privileges, or put a user in the Admin group, that user has full access to everything in the repository. Do not make any ACLs for anyone with Admin role, privilege, or group. If you need to exclude a user from certain files, assign that user another role without any use of the Admin privilege, role or group.

---

### 5.1.1 Special Permissions

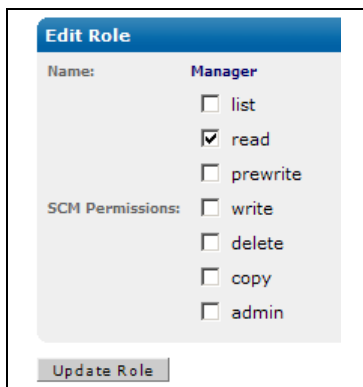
Special consideration should apply for list and read access rules. Unlike write operations, the read and list operations can traverse directory hierarchy. Therefore it makes sense to always allow/deny read and list privileges on all files under a directory. This can be done by specifying a wild-card pattern, for example: `allow read from /svnroot/trunk/module1|/svnroot/trunk/module1/.`

## 5.1.2 Creating New Roles

In the Security tab, select **Create Role**. Enter a name for the role and select the Subversion permissions you would like this role to have. Any user you assign to this role has the permissions you specify for this role.

## 5.1.3 Editing Existing Roles

Select **List Roles**: the defined roles display. Select the name of the role you wish to edit. The Edit Role page displays, listing all possible privileges. The role's existing privileges are checked.



Make any changes, and click **Update Role**. Any user assigned to that role, both for current and future assignments, has these same privileges.

## 5.1.4 Deleting Roles

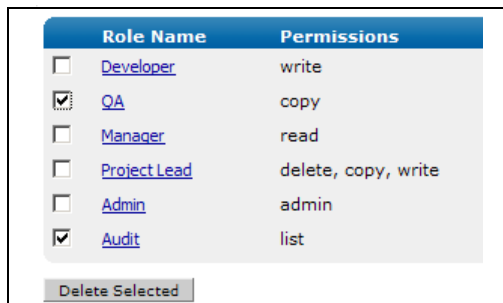
Select **List Roles**. Use the checkboxes to mark the roles for deletion. Select **Delete Selected**. The role is deleted throughout Access Control, even if users are assigned to that role.

**WARNING:**

---

Think carefully when deleting roles. If you delete a role, make sure no user is assigned to that role before you delete it.

---



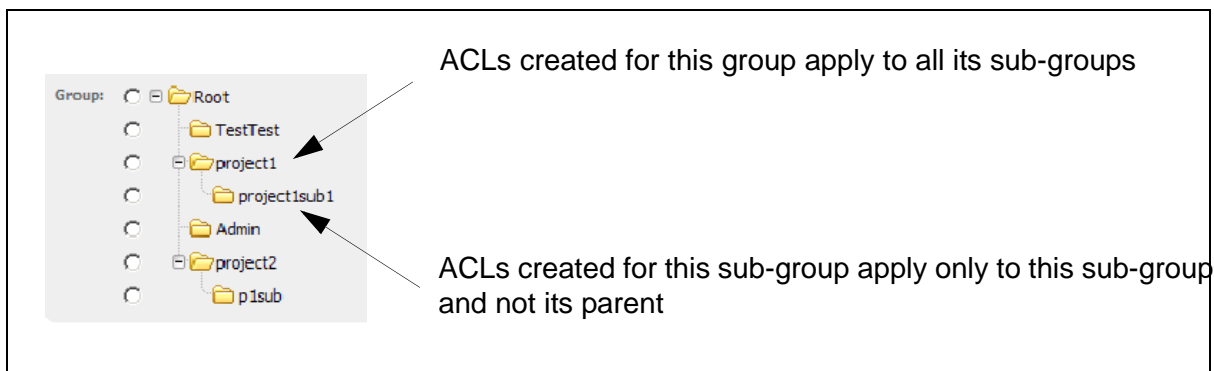
| Role Name                                                 | Permissions         |
|-----------------------------------------------------------|---------------------|
| <input type="checkbox"/> <a href="#">Developer</a>        | write               |
| <input checked="" type="checkbox"/> <a href="#">QA</a>    | copy                |
| <input type="checkbox"/> <a href="#">Manager</a>          | read                |
| <input type="checkbox"/> <a href="#">Project Lead</a>     | delete, copy, write |
| <input type="checkbox"/> <a href="#">Admin</a>            | admin               |
| <input checked="" type="checkbox"/> <a href="#">Audit</a> | list                |

## 5.2 Managing Groups

Creating groups allows you to manage projects, providing a convenient way of organizing many users into a related category for controlling access. You assign each group to a set of files, a directory hierarchy, or to individual directories, to either allow or deny access to specified files and directories.

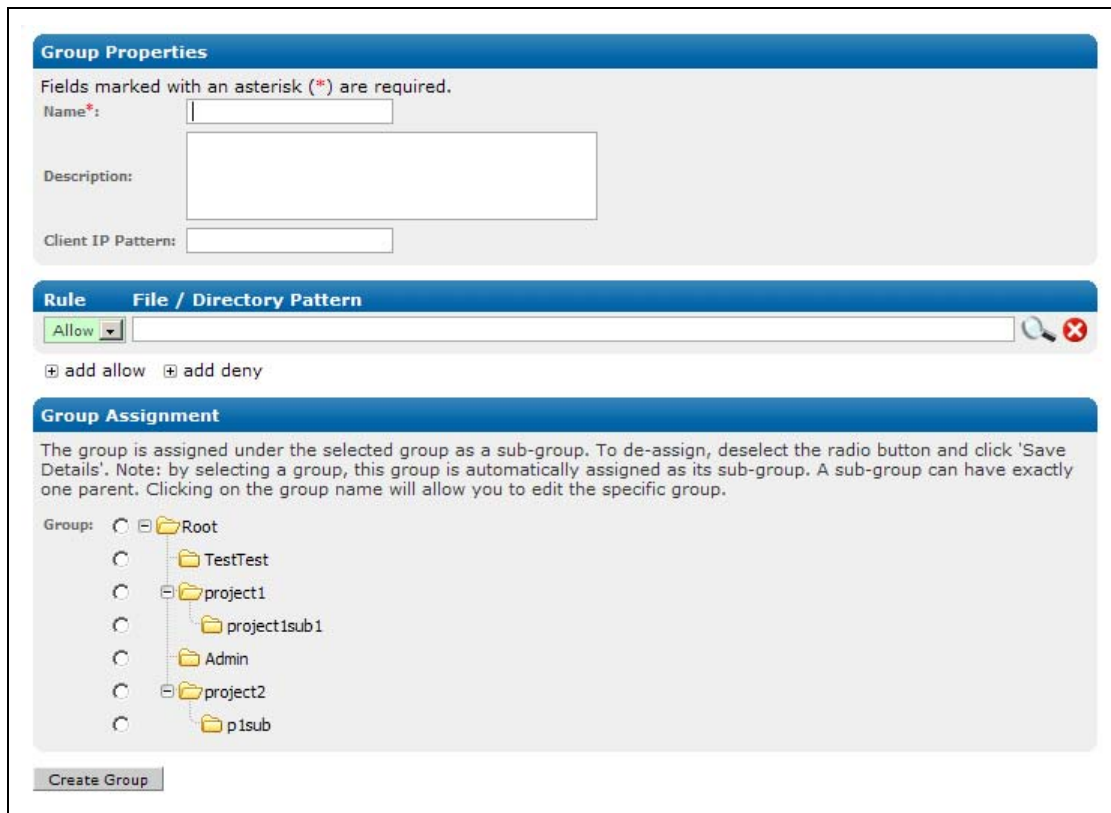
You can create and delete groups, associate files, directories and modules to a group, add to and remove users from a group, and perform bulk imports of existing groups. You can also restrict access to a group by client IP address.

Groups are hierarchical, with a parent-child association between a group and a sub-group.



## 5.2.1 Creating New Groups

To add a new group, select **Create Group**. The Group Properties page appears.



**Group Properties**

Fields marked with an asterisk (\*) are required.

Name\*:

Description:

Client IP Pattern:

---

**Rule** File / Directory Pattern

Allow

+ add allow + add deny

---

**Group Assignment**

The group is assigned under the selected group as a sub-group. To de-assign, deselect the radio button and click 'Save Details'. Note: by selecting a group, this group is automatically assigned as its sub-group. A sub-group can have exactly one parent. Clicking on the group name will allow you to edit the specific group.

Group:  Root

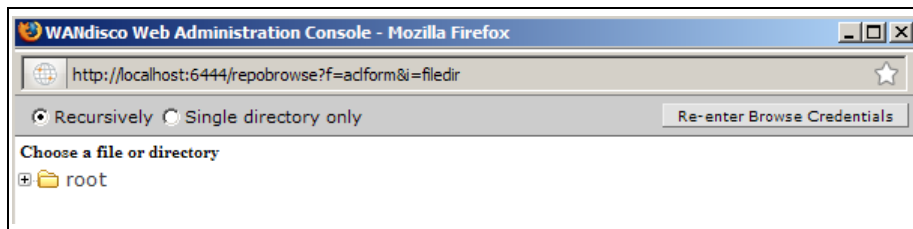
- TestTest
- project1
  - project1sub1
- Admin
- project2
  - p1sub

The name can contain any character, including white space, except the underscore character. The group name is the primary key into the group database, therefore it cannot be changed once it is created. Enter relevant text in the description field. Access Control automatically tracks the creation and modification time on the groups, which you can see in `groups-reports.txt` in **Log Viewer**.

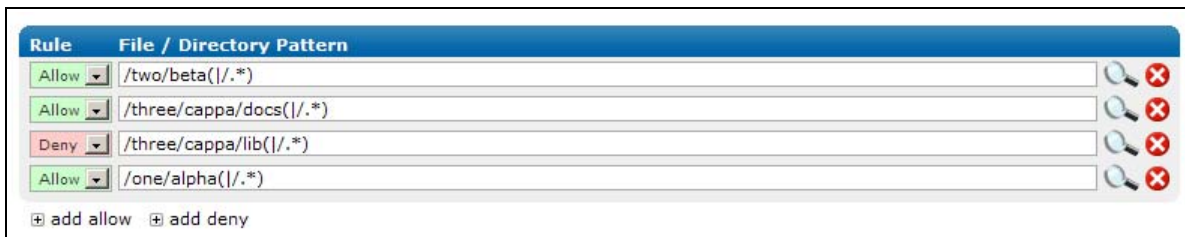
You can optionally create this group for a specific client IP pattern. If you do enter an IP pattern in the Client IP Pattern field, no other client IPs are allowed unless you create specific ACLs for those other client IP addresses. You must use regular expressions.

### 5.2.1.1 Defining Group Rules

The **Rule** section allows you to define the files and directories for this group. Select **add allow** or **add deny**, and browse to the file or directory. You can identify a file or directory, and use the radio buttons to check either Recursively or Single directory only.



Add as many entries as necessary for this group, ensuring that all allows and denies for all files and directories are accounted for.



### 5.2.2 Creating a Sub-Group

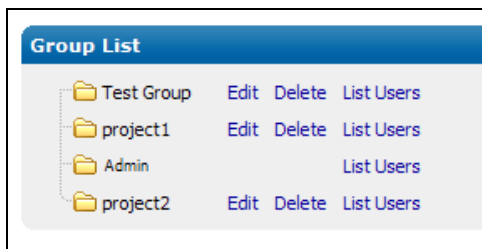
A group inherits all of the resources and privileges of its sub-groups.

Follow these steps to create a sub-group.

- Step 1 Make the sub-group as you would a group.
- Step 2 Go to **List Groups**.
- Step 3 Click **Edit** for the sub-group. The Group Properties page appears.
- Step 4 In the **Group Assignment** section, check the radio button of the sub-group's parent.
- Step 5 Click **Save Changes**.
- Step 6 Go back to **List Groups** and verify the correct structure.

### 5.2.3 Deleting a Group

To delete a group, click **List Groups**.



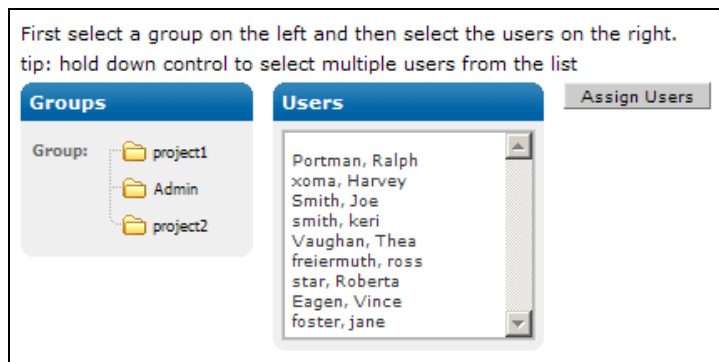
Click **Delete**. You are asked to confirm your action.

When you delete a group, the association between the group and any users who belonged to that group is broken. The associations between any sub-groups and users are also deleted.

If you want to keep a sub-group, first select a new parent for that sub-group, and then delete the old parent group. The sub-group then does not get deleted.

### 5.2.4 Assigning Users to a Group

To add users to a group, click **Assign Users**.



Select a group on the left. The list of users on the right updates to reflect potential new members for the group you selected. Users already in the group are excluded from the Users list.

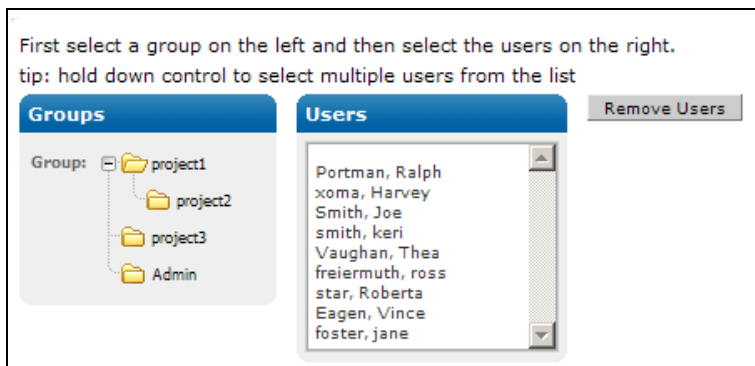
If a user belongs to a parent group, they automatically belong to any sub-groups underneath it, even though the list does not reflect that. However, a user can belong to a sub-group and not belong to the parent group.

Select the users to add to the group. To add several users at once, hold down the Control key while you click on your selections.



## 5.2.5 Deleting Users from a Group

To delete users from a group, click **Remove Users**.



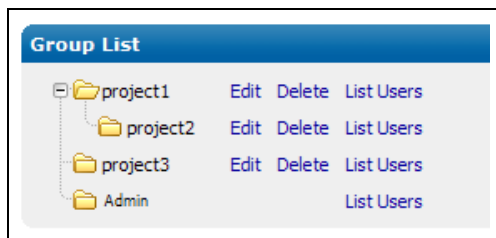
Select a group on the left. The list of users on the right updates to reflect that group's users.

Select the users to remove from the group. To remove many users at once, hold down the Control key while you click on your selections. Click **Remove Users**.

If a user belongs to a parent group, they automatically belong to any sub-groups underneath it; however, the screen does not reflect this. If a user is removed from a parent group, they are also removed from any sub-groups.

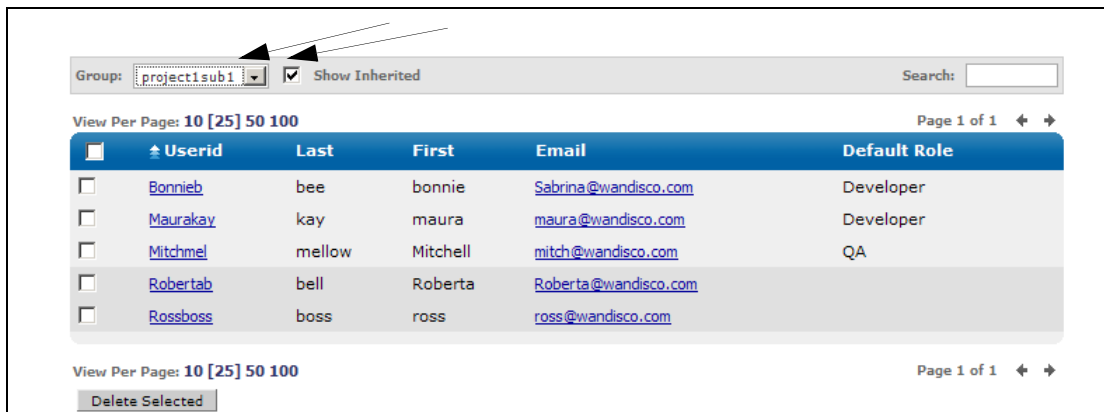
## 5.2.6 Who Is In a Group?

To view a list of which users belong to which groups, click **List Groups**.



All the groups are displayed. Click **List Users**. All the users in that group are displayed.

To view users who are explicitly members of this sub-group and those members inherited from any parent groups, check the **Show Inherited** checkbox. Use the Group drop-down list to view the users belonging to another group.

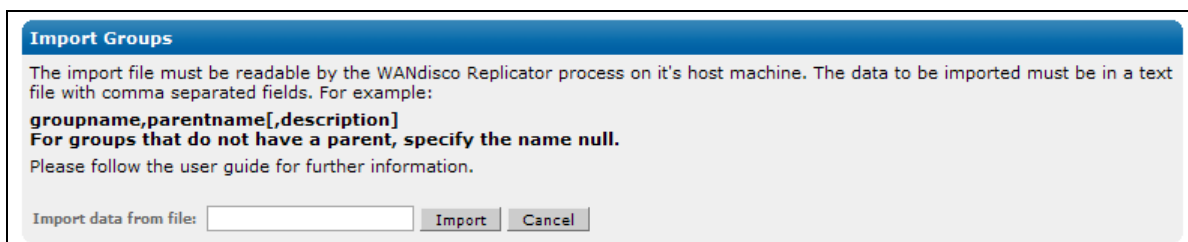


The userids are linked to the User Properties page, in case you need to edit a user. You can also edit and delete groups from this page.

## 5.2.7 Importing Existing Groups

You may have groups already set up outside Access Control. If so, you can import them using the **Import Groups** command, in a comma separated text file, of the format `groupname,parent-name<,description>`.

Type in the pathname to the file, and click **Import**. The new groups are added to the existing groups. Define the resources for this group or subgroup, and assign users.



## 5.3 Managing Users

The User Administration commands allow you to create and delete users, assign a user roles and groups, and search users by several criteria.

On installation, if you selected to have WANdisco control the Subversion password file, any user you enter can use Subversion.

If WANdisco is not managing the Subversion password file, and is entered in WANdisco but has not been registered in Subversion, and he or she tries to access Subversion, they would see an `Access Denied` error message in their client.

You can check if WANdisco is managing the password file by looking at the **SVN Password file** value on the Proxy tab. If a path is listed, WANdisco is managing the password file. Otherwise, you see a `not managed by WANdisco` message.

If you selected the wrong choice during installation, you can still have WANdisco manage the password file. In `prefs.xml`, change `false` to `true`, and specify the absolute pathname to password file. See the procedure [8.2, Changing the prefs.xml File](#).

```
<SVNProxy>
  ...
  <svnautoupdate>true</svnautoupdate>
  <svnpasswdfile><C:\svn-repository\dav-auth</svnpasswdfile>
  ...
</SVNProxy>
```

### 5.3.1 Creating or Removing Users

To add a new user, click on **Create User** in the Security tab and specify a (Subversion) username. You can optionally specify an email address for the user. A default role is required to define the user's privileges. You can assign the user to groups here, or through the Group Administration commands.

To remove users, click **List Users**. Select the users you want to delete with the checkbox on the left and click **Delete Selected**.

### 5.3.2 Assigning Users to a Sub-Group

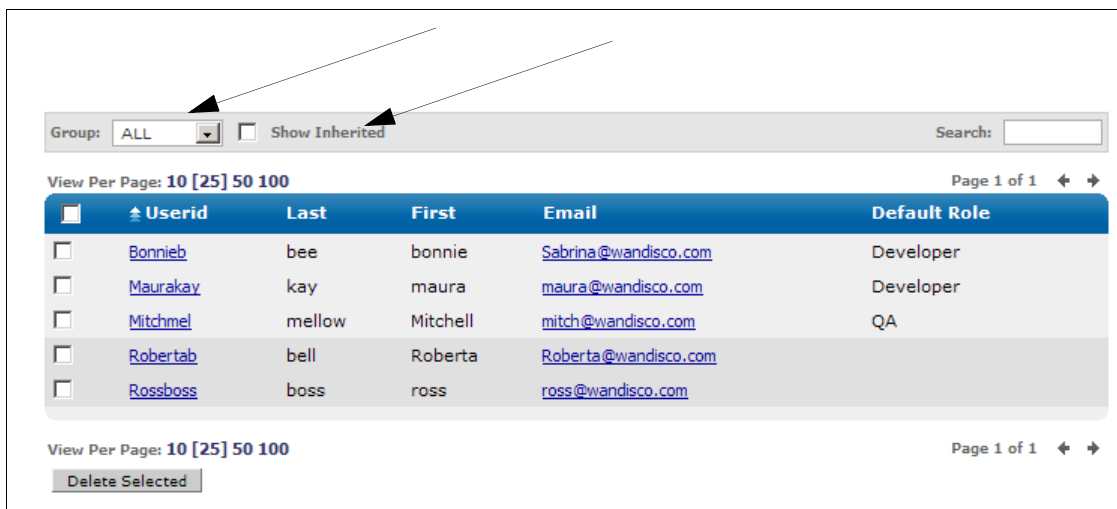
You can assign a user to any number of groups with the **Assign Users** command. Note by selecting a group, the user is automatically assigned to the group and all its sub-groups. To unassign, check the checkbox and click **Save Details**.

### 5.3.3 Listing and Searching for Users

To get a list of all the registered users, click on the **List Users** command. The User List page shows all users by default. The page size is set to show 25 users per page, but you can change that by selecting **View Per Page** on top of the user list. Arrows at the right corner allow you to page to the next or previous page. Use the **Search** box: start typing a user's first or last name, and an incremental search starts. Return to the full list by clearing the **Search** box.

All the columns in the user list are enabled for sorting. Clicking on the column header lets you sort in ascending or descending order. The sortable columns include: Userid, last name, first name, email, and role.

To restrict the list by group, select a group name from the Group drop-down list. To see users in the selected group, as well as all the ancestor groups, check the **Show Inherited** checkbox.



You can click on the user id hyper link to edit the user. You can also delete as many users as you like. Delete all users by checking the checkbox in the table header, and then click the **Delete Selected** button.

## 5.3.4 Importing Users

You can import an existing list of users with the **Import Users** command. The import file must be a comma delimited text file, of the format `username,lastname,first-name,email[,group1[,group2,...groupN]]`.

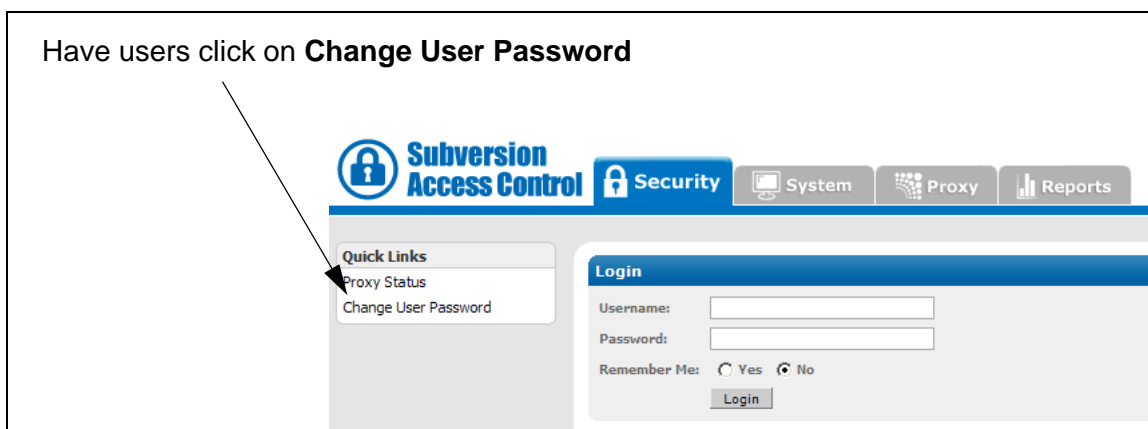
If WANdisco is controlling the Subversion password file, user passwords are changed to their email addresses. WANdisco recommends notifying users to change their Subversion password, as described in the next section.

### 5.3.4.1 Having Subversion Users Change Their Passwords

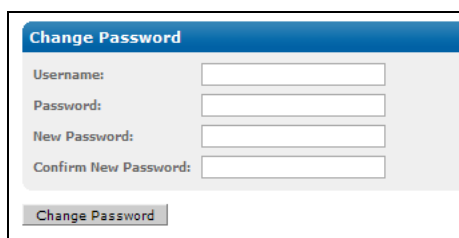
Users can change Subversion passwords in WANdisco without logging in to WANdisco. Have the users go to

`http://localhost:6444/`

The Admin Console appears. Have the users click on **Change User Password**.



The Change Password box appears. Users can enter their Subversion username, and their password (which is now their email address). Have them enter a new password and confirm it, then click **Change Password**. The users have successfully changed their passwords.



## 6 About Access Control Lists

Most people find that managing users' roles and groups offer enough access control. However, Access Control allows you to have very specific control of users through the use of Access Control Lists (ACLs).

### 6.1 How WANdisco Enforces Rules

When a user tries to execute a Subversion command, Access Control's ACL engine always follows the same process to make an allow or deny decision.

First, the ACL engine checks if a user is registered or licensed in the WANdisco user database. If the user is not registered or licensed, the user is denied access.

In order for a rule to be matched, the ACL engine verifies that a user's name or the group(s) a user belongs to, IP address and file/directory matches the patterns specified in the ACLs.

Rules applicable to a specific user override the rules applicable to a group.

User's access rights	Group's access rights	WANdisco allows or denies?
none specified	allowed	allowed
none specified	multiple groups, any of which is allowed	allowed
none specified	multiple groups, any of which is denied	denied
denied	multiple groups, any of which is allowed	denied
allowed	denied	allowed
denied	denied	denied

WANdisco allows you to automatically edit multiple rules. When you submit changes to ACLs, WANdisco guarantees either all the rules are updated or none at all.

When setting up a rule on a specific directory, note that the directory name is treated as a regular expression pattern. For example, if you want to allow write access to all the files under a directory `/svnroot/trunk/docs`, you need to specify one of the following patterns:

```
/svnroot/trunk/docs | /svnroot/trunk/docs/. *
```

or

```
/svnroot/trunk/docs. *
```

The first pattern allows write into the directory (to create new files or directories) as well as all files under the `.../docs/` subdirectory. The second pattern allows access to all files and subdirectories that match `/svnroot/trunk/docs`, including, `/svnroot/trunk/docs`, `/svnroot/trunk/docsmaker`, `/svnroot/trunk/docs2`, etc.

Special considerations should apply for list and read access rules. Unlike write operations, the read and list operations can traverse directory hierarchy. Therefore it makes sense to always allow or deny read and list privileges on all files under a directory. This can be done by specifying a wild-card pattern, for example:

```
allow read from /svnroot/trunk/module1|/svnroot/trunk/module1/*.
```

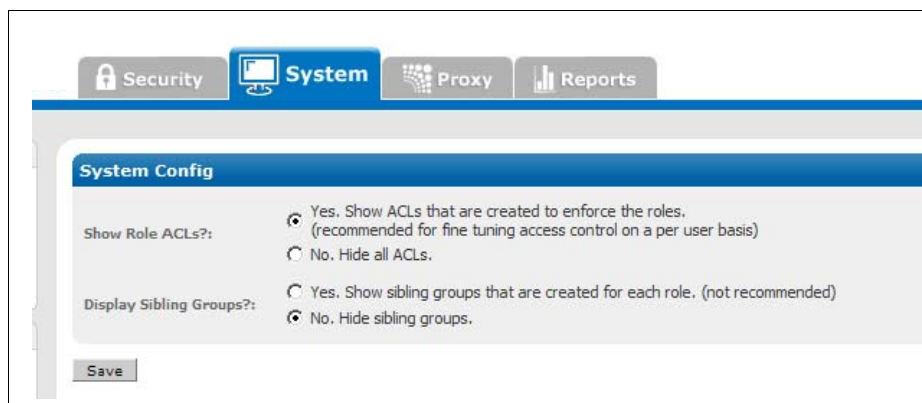
To use the copy privilege, specify it on the source directory. It allows a user to copy from a given directory. Make sure you enable the write privilege on the parent directory of the intended destination. Granting write privilege does not imply the user has delete or copy privilege. This allows the administrator to control who can create tags or branches and who can delete version controlled files. For example, to allow copy from `/trunk` to `/tags/re11`, you create two access rules:

- Allow Copy from `/trunk.*`
- Allow Write to `/tags/re11`

## 6.2 Toggling the ACL Display

You can toggle the display of role ACLs. The default is off, which is recommended when you are just setting up the roles and groups.

Go to the System page, and click **System Config**. Select the **Yes** or **No** radio button for Show ACLs?



When toggled on, you see any ACLs created by roles and groups listed on the Group Properties page, shown in the next illustration.

**Group Properties**

Fields marked with an asterisk (\*) are required.

Name\*: **TestTest**  
 Created: **Monday, October 20, 2008 1:16:30 PM**  
 Modified: **Monday, October 20, 2008 1:16:30 PM**

Description:

Client IP Pattern:

---

**Rule File / Directory Pattern**

Allow	/data/project1(/.*)	🔍 ✖
Allow	/data/project1(/.*)	🔍 ✖
Allow	/data/to_thea.txt	🔍 ✖
Deny	/data/project2/test.txt	🔍 ✖

⊞ add allow   ⊞ add deny

---

**Group Assignment**

The group is assigned under the selected group as a sub-group. To de-assign, deselect the radio button and click 'Save Details'. Note: by selecting a group, this group is automatically assigned as its sub-group. A sub-group can have exactly one parent. Clicking on the group name will allow you to edit the specific group.

Group:  Root

- p1sub
- TestTest
- project1
- Admin
- project2

---

**Membership**

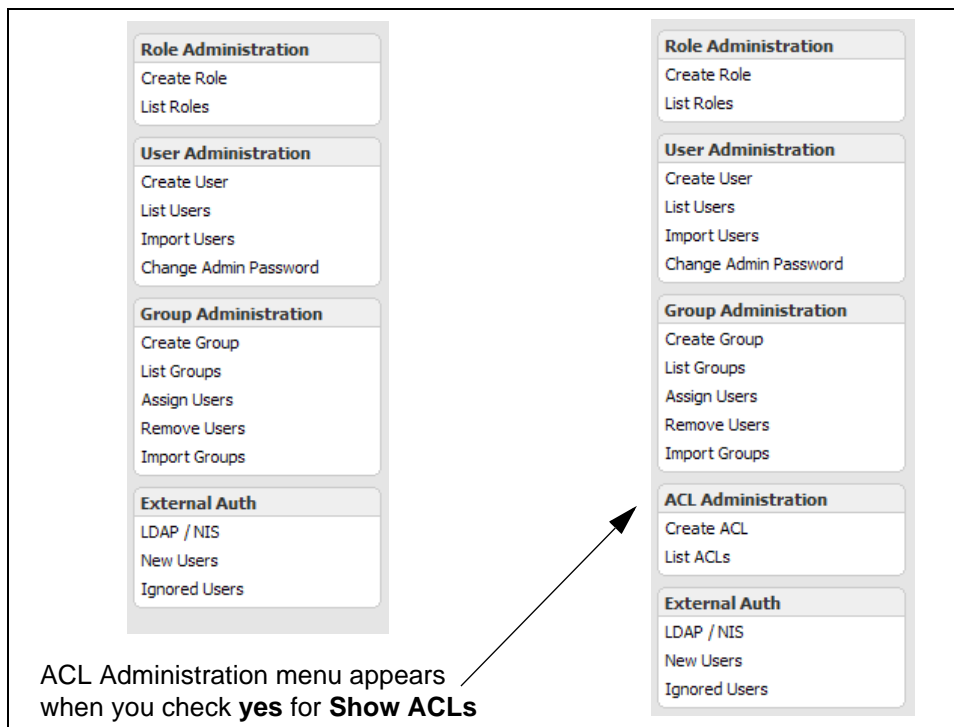
[List Users in this group](#)

Id	Rule	Privilege	User / Group Pattern	Is Group?	IP Pattern	File / Dir Pattern
0	Deny	from user role	TestTest	true	192.168.1.*	/data/project2/test.txt
1	Allow	read	.*	true	.*	/data
2	Allow	prewrite	.*	true	.*	/data
3	Allow	read	.*	true	192.168.1.*	/data
4	Allow	prewrite	.*	true	192.168.1.*	/data
5	Allow	from user role	TestTest	true	192.168.1.*	/data/project1(/.*)
6	Allow	from user role	TestTest	true	192.168.1.*	/data/project1(/.*)
7	Allow	from user role	TestTest	true	192.168.1.*	/data/to_thea.txt

ACLs created from roles and groups



toggling the ACL display also displays the available ACL Administration commands in the Security tab.



### 6.3 Creating ACLs

To create ACLs, go to the Security tab and click **Create ACL**. If you are creating multiple ACLs, click on **List ACLs**. Refer to the field descriptions in [ACL Administration](#) in Chapter 4, [Using the Admin Console](#).

### 6.4 Toggling the Use of Access Control Lists

The following properties in the `prefs.xml` file can be used to control the ACL engine.

```
<Security>
  <AccessControl>
    <Enable>true</Enable>
    <Replicate>true</Replicate>
    <ClientTimeout>15s</ClientTimeout>
  </AccessControl>
</Security>
```

By default, Access Control has access control enabled. To turn it off, add the lines to `prefs.xml` and set `Enable` to `false`.

## 7 About Audit Reports

---

Access Control logs any Subversion user access (allowed or denied) in an audit trail file. WANdisco supplies a standard report, Users and Groups, but recommends you import the data into a database such as MySQL, so that you can make complex queries. WANdisco offers three such reports when set up with a database: Transaction History, Access Violation Report, and File.

To set up the more detailed reports, you need to:

- install php and the appropriate Apache module for your system
- install a database such as MySQL
- make sure you have the `svn-security/config/reports.tar` file, which contains WANdisco-supplied php scripts. This is part of the standard distribution.

To ensure no audit records are lost, WANdisco recommends you schedule a job (using `cron`, for example) to import the audit records into a database periodically.

### 7.1 Setting Up the Reports

Use this procedure to set up the reports. WANdisco does not automatically import data into the database. You can do this manually or set up a `cron` job.

- Step 1 Download and install a database such as MySQL.
- Step 2 Download and install php.
- Step 3 Set up a user in the database. Grant that user all privileges to manage the database `wd_audit_db` (created in the next step).
- Step 4 Create a database called `wd_audit_db` to manage the database. The database schema is created upon the import tool's first population, which you perform at a later step.
- Step 5 Decompress the php scripts at `svn-security/config/reports.tar`.
- Step 6 Edit the `reports/config.php` to point to the database you just created. Modify the `config.php` file to update the server, username and password entries along with the `scm` type, which is `svn`.
- Step 7 Edit the `importauditdb` script to match the changes to `config.php`: `dbhost`, `dbuser`, and `dbpass`. It is recommended to not use the default user, `root`.

- Step 8 Update your Apache `httpd.conf` file to point to the scripts. Make sure to replace `/home/wandisco/reports` with your installation directory. You may also want to rename the `/reports/` alias (e.g. `/wandisco_reports`). For example,

```
Alias /reports/ "/home/wandisco/reports/"
<Directory "/home/wandisco/reports">
Options Indexes MultiViews
AllowOverride None
Order allow,deny
Allow from all
</Directory>
```

- Step 9 Restart Apache.
- Step 10 Run the import tool. See [7.1.1, Using the Import Tool](#).
- Step 11 You can now run the reports. See [7.3, Running a Report](#).

## 7.1.1 Using the Import Tool

The import tool requires `Perl::DBI` module to be installed. Please run `svn-security/bin/checkdbi` to verify that the module is installed, and the correct database driver is available on your system.

The import tool is called `importauditdb`, and its usage is as follows:

```
perl importauditdb -host dbserver -user dbuser -pass dbpassword -f ../
audit/audit-trail.0
```

Here is an example of how to use the import command:

```
[admin@smp1 ~/svn-replicator]$ bin/importauditdb -h
Usage:
  importauditdb [-host <db-host>] [-port <db-port>] [-user <db user>]
                [-pass <db user password>] [-db <database to use>]
                -f file-pattern1 file-pattern2 .. file-pattern-N

Defaults:
host : localhost
port : Default DB Port
user : root
password : empty
Database : wd_audit_db
```

### NOTE:

---

Before using import, you must create a database on the database server.

---

The import tool automatically creates the table schema in that database, the first time it runs. The import tool uses standard SQL syntax, and makes use of a system function `FROM_UNIXTIME`. Please ensure your database version supports it. MySQL and Microsoft SQLServer both support this function. Here is an example of how you would import a file:

```
perl importauditdb -host dbserver -user dbuser -pass dbpassword -f ../
audit/audit-trail.0
```

The `audit-trail.0` file is located in the `svn-security/audit` directory. The file has a complete history of all Subversion actions, listed in the following format:

```
# Column syntax -
# 0 seq | 1 time | 2 txid | 3 cmd | 4 user | 5 ipaddress | 6 access |
# 7 dir | 8 file | 9 rev
```

The columns are described in this table:

Column No.	Description
0	Record Sequence Number
1	UNIX Timestamp
2	Transaction Id
3	Subversion Command Name
4	Subversion User id
5	IP Address of User
6	Access Decision (Allow or Deny)
7	Subversion Directory being accessed
8	Subversion File being accessed
9	User's File Revision

## 7.2 Configuring Audit Properties

Auditing is controlled in the `prefs.xml` file. By default, Access Control enables auditing. You can turn it off by setting the `Disable` element to `true`.

```
<Audit>
  <MaxFileSize>10485760</MaxFileSize>
  <MaxFileCount>10</MaxFileCount>
  <Disable>false</Disable>    <!-- this is the default -->
</Audit>
```

By default, Access Control automatically rotates the files up to 10 times when they get to 10 megabytes. You can change these defaults in the `prefs.xml` file. The `MaxFileSize` element specifies a size in bytes, and the `MaxFileCount` element specifies how many files to rotate before recycling the files.

To create audit files in a different directory, create a symbolic link (`svn-security/audit`) to another directory.

**NOTE:**

---

You do not want to lose any audit history. Make sure that any interval you schedule to import the files into a database is short enough (not longer) so that all files in the `MaxFileCount` element are captured (and not overwritten).

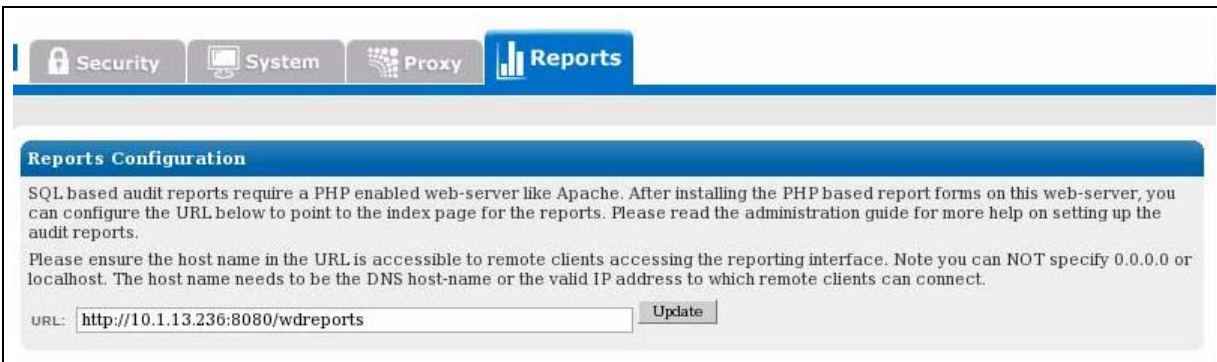
---

## 7.3 Running a Report

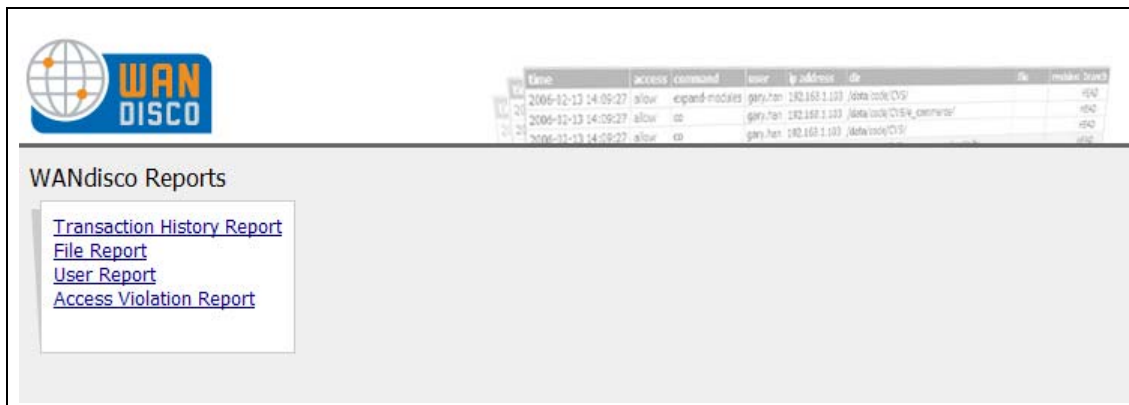
- Step 1 Configure the report URI. Go to the Reports tab in the Admin Console. Click **Configure URI**. Enter in the IP address of the <reports apache server>:8080/<reports direcotry>. For example,

`http://10.1.13.236:8080/wdreports`

Click **Update**.



- Step 2 Go to that URL.



- Step 3 Select **File Report** from the main menu.
- Step 4 Enter the criteria for the report. For example, select a user from the drop-down, specify an access level or a Subversion command to filter the results. Note: use % for wildcards.
- Step 5 Click **Run Report**.

## 7.3.1 Report Types

Report Name	Description
Transaction History	Shows all transactions against Subversion.
File	List file access and filter by parameters such as: date, access, command, user, ip address, directory, filename, revision or branch.
User	Show Subversion allowed / denied access per user.
Access Violation	Display all denied access to Subversion.

### 7.3.1.1 The User Report

**User Report**

Report Timestamp: **Oct 20, 2008 02:09:14**

From Date: **10/17/2008**

To Date: **10/20/2008**

From:  To:

Filename:

time	access	command	user	ip address	file path	revision
2008-10-17 10:26:39	deny	OPTIONS	tvaughan	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data	
2008-10-17 10:27:16	allow	REPORT	Bonnieb	192.168.1.184	/data	
2008-10-17 10:27:16	allow	OPTIONS	Bonnieb	192.168.1.184	/data/project1	

[Return to Reports Home](#) [previous](#) [next](#)

### 7.3.1.2 The Transaction History Report

Transaction History Report

Report Timestamp: **Oct 20, 2008 02:11:05**

From:  To:

Filename:

time	access	command	user	ip address	file path	revision
2008-06-18 14:31:06	allow	list	veagen	0:0:0:0:0:0:1	/project	
2008-06-18 14:31:09	allow	list	veagen	0:0:0:0:0:0:1	/	
2008-06-18 14:31:09	allow	read	veagen	0:0:0:0:0:0:1	/	
2008-06-18 14:31:19	allow	list	veagen	0:0:0:0:0:0:1	/trunk/project	
2008-06-18 14:31:19	allow	read	veagen	0:0:0:0:0:0:1	/trunk/project	
2008-06-18 14:31:30	allow	write	veagen	0:0:0:0:0:0:1	/trunk/project/build.xml	
2008-10-17 10:26:39	deny	OPTIONS	tvaughan	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	OPTIONS	Bonnieb	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data/project1	
2008-10-17 10:27:16	allow	PROPFIND	Bonnieb	192.168.1.184	/data	

[Return to Reports Home](#) [previous](#) [next](#)

### 7.3.1.3 The Access Violation Report

Access Violation Report

Report Timestamp: **Oct 20, 2008 02:12:40**

From Date: **10/17/2008**

To Date: **10/20/2008**

Access: **deny**

From:  To:

Filename:

time	access	command	user	ip address	file path	revision
2008-10-17 10:26:39	deny	OPTIONS	tvaughan	192.168.1.184	/data/project1	
2008-10-17 10:32:36	deny	PUT	Mitchmel	192.168.1.184	/data/project1/trunk/readme.txt	

[Return to Reports Home](#) [previous](#) [next](#)

### 7.3.1.4 The File Report

File Report

Report Timestamp: **Oct 20, 2008 02:14:54**

From Date: **10/17/2008**

To Date: **10/20/2008**

Access: **deny**

User: **Mitchmel**

From:  To:

Filename:

time	access	command	user	ip address	file path	revision
No results found.						

[Return to Reports Home](#) [previous](#) [next](#)



## 8 Procedures

---

### 8.1 Preventing Subversion Users From Making Transactions

**NOTE:**

---

This procedure involves taking Subversion offline. Please follow your company procedures about notifying Subversion users of down time.

---

- Step 1 After notifying your users of the downtime, navigate to the Proxy tab.
- Step 2 Stop Access Control. Click **Stop Proxy**.
- Step 3 Verify Access Control is shut down. On the Proxy tab, the value for the `listening` field should be `no`.
- Step 4 To start Access Control, click **Start Proxy**.

### 8.2 Changing the `prefs.xml` File

The `prefs.xml` file is located in `svn-security/config`. This procedure involves stopping user access to Subversion. Please follow your company guidelines for notifying users of downtime.

- Step 1 Back up the existing `prefs.xml` file.
- Step 2 Make any modifications to the file.
- Step 3 Notify users of Subversion downtime.
- Step 4 Stop Access Control. Go to the Proxy tab, and select **Shut Down Node**.
- Step 5 Restart Access Control. At the command line, type

```
svn-security/bin/svnsecurityagent
```

Access Control restarts with the new `prefs.xml` file.

## 8.3 Verifying That Access Control is Working

Check if Access Control is working by verifying there are commit transactions posted to the log file `svn-security/logs/svnProxyServer-prefs.log`.

```
INFO: [listen-1] Listening on port : 0.0.0.0/0.0.0.0:6445
1219077847375 org.nirala.communication.transport.DConeNet.AsyncConnector
makeConnection
INFO: [main] Connection request to Node Id = c66b6db9-6a50-11dd-8675-
001aa036534c, host = 192.168.1.15, port = 6666, timed out in 500ms
1219077847875 org.nirala.communication.transport.DConeNet.AsyncConnector
makeConnection
INFO: [main] Connection request to DFTPEndpoint - Node Id =
192.168.1.156666, host = 192.168.1.15, port = 6666, timed out in 500ms
1219077848578 org.nirala.admin.DiskMon start
INFO: [main] Diskmon is monitoring C:\Thursday\svn-ha\systemdb every
15min
1219077849000 org.nirala.communication.transport.DConeNet.ListenReactor
setupListener
INFO: [listen-1] Listening on port : 0.0.0.0/0.0.0.0:80
1219077849000 org.nirala.communication.transport.svnproxy.ProxyServer
onStartedProxyListen
INFO: [main] SVN Proxy listener is now turned ON at port :80
1219077853765 org.nirala.communication.transport.DConeNet.Listen-
Stage$TCPStopListening onStop
INFO: [listen-1] Host: 0.0.0.0, Port: 2403 Stopped Listening.
1219077853765 org.nirala.communication.transport.svnproxy.ProxyServer
onStopProxyListener
INFO: [p-queue-1] SVN Proxy listener is now turned OFF at port :80
1219077872328 org.nirala.communication.transport.DConeNet.ListenReactor
setupListener
INFO: [listen-1] Listening on port : 0.0.0.0/0.0.0.0:80
1219077872328 org.nirala.communication.transport.svnproxy.ProxyServer
onStartedProxyListen
INFO: [mqueue-1] SVN Proxy listener is now turned ON at port :80
```

## 8.4 Installing a .jar File Patch

Follow these steps to install a `svn-security.jar` patch to an existing Access Control installation.

**NOTE:**

---

Always read the `readme` file first.

---

- Step 1 Download the `svn-security.jar` file.
- Step 2 Verify the md5 checksum.
- Step 3 Move the existing jar file to a back up directory. (All jar files in the `/lib` directories are in the WANdisco `CLASSPATH`.)
- Step 4 Stop Access Control. See [8.1, Preventing Subversion Users From Making Transactions](#).
- Step 5 Copy the new jar file to the `lib` directory.
- Step 6 Restart the proxy.
- Step 7 Confirm the upgrade by checking the lower right corner of Admin Console for the newer version, and check the log file under `svn-security/logs` for the start header with the new version.

## 8.5 Setting WANdisco to Start Up on System Boot

You can easily have WANdisco start up on system boot. To start up on boot, edit the `init.d` scripts. You must make modifications based on your operating system. Edit the script accordingly.

For instance, here is an `/etc/init.d/svnsecurity` script for Gentoo Linux.

```
#!/sbin/runscript
#
# Gentoo Linux dist compatible rc script for
# starting/stopping svnsecurity
#
# Copyright WANdisco
#

REP_HOME="/home/admin0/svn-security"
REP_OPTS="-wdog -email admins@foo.com"
export JAVA_HOME="/export/share/apps/jdks/1.5.0"
USER="admin0"

pidfile="my.pid"

depend() {
    need net
}

checkconfig() {
    if [ ! -f ${REP_HOME}/bin/svnsecurity ]; then
        eerror "No ${REP_HOME}/bin/svnsecurity present"
        return 1
    fi
    prog=${"svnsecurity"}
}

start() {
    checkconfig || return 1
    ebegin "Starting $prog:"
    ulimit -S -c 0 >/dev/null 2>&1
    ulimit -n 65000 >/dev/null 2>&1
    RETVAL=0
    start-stop-daemon --start --quiet -u ${USER} --chuid ${USER} --exec
    ${REP_HOME}/bin/svnsecurity -- ${REP_OPTS}
    RETVAL=$?

    if [ "$RETVAL" -gt 0 ]; then
        eend $RETVAL "Failed to bring up svnsecurity"
        return $RETVAL
    fi
    eend $RETVAL
}
```

```
stop() {
checkconfig || return 1
ebegin "Shutting down $prog:"
su ${USER} -c \"${REP_HOME}/bin/shutdown\" >/dev/null 2>&1
start-stop-daemon --stop --quiet -u ${USER} --pidfile ${REP_HOME}/logs/
${pidfile}
RETVAL=$?
if [ "$RETVAL" -gt 0 ]; then
eend $RETVAL "Failed to shutdown svnsecurity"
return $RETVAL
fi
eend $RETVAL
}
```

## 8.6 Setting WANdisco Up as a Windows Service

To set WANdisco to run as a Windows service, perform the following command at the command prompt:

```
sc create SVN-security binpath= C:\perl\bin\perl.exe -x -S C:\svn-secu-
rity\bin\svnsecurity start= auto
```

Substitute the path for Perl in your environment, and give a different path to the Subversion security perl script, depending on where that was installed. You may want to also set `type= share`. The MicroSoft knowledge base article (<http://support.microsoft.com/kb/251192>) indicates that that is the default, but the `sc.exe help for create` indicates that `type= own` is the default. Note that there is a space between the equals sign, `=`, and the parameter's value.

The Services Control Panel indicates that the service has not started, because our Perl script is currently not exiting because the watchdog is running to restart WANdisco. This is actually fine, because the Perl script really takes over.

## 8.7 Changing SVN Port on Unix Flavor

You can change the port in the `httpd.conf` configuration file in the Apache server. Please see the `Listen` directive, discussed in the article at <http://httpd.apache.org/docs/2.2/bind.html>.

Here is a snippet of an `httpd.conf` file:

```
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses (0.0.0.0)
#
#Listen 12.34.56.78:80
Listen 8080
```

With this configuration, Apache server listens on port 8080 instead of default port 80.

## 9 Troubleshooting

---

### 9.1 How Do I Get WANdisco Support?

Before opening a ticket or submitting a new issue, always search the Knowledge base on [http://www.wandisco.com/php/support\\_login.php](http://www.wandisco.com/php/support_login.php).

If you want to open a ticket, you can do so at that URL.

### 9.2 Apache and SVNKit

There are some known behaviors with connection pooling when using Apache and SVNKit. Adding a WANdisco product does not change the behavior of these issues, but you may need to revisit them after adding WANdisco.

WANdisco recommends using JavaHL with Eclipse IDE, which does not use connection pooling, and thereby eliminates any problems.

#### 9.2.1 SVNKit and Connection Pooling

SVNKit uses connection pooling. For a given client, SVNKit opens two connections and keeps them open for later use. On a system with a heavy load and numerous clients, this can cause performance degradation. An open connection consumes an Apache worker thread, and with many clients and connection pooling, Apache may run out of worker threads. Apache provides various tuning parameters to optimize connection pooling but still release the unused connections. The tunable parameters are Timeout, KeepAliveTimeout, MaxKeepAliveRequests, and KeepAlive. Please refer to the Apache configuration documentation for further details at <http://httpd.apache.org/docs/2.2/mod/core.html>.

#### 9.2.2 Tuning Values to Optimize Your Configuration

Apache has two timeout configurations: Timeout and KeepAliveTimeout. In general, the Timeout value should be higher than the value for KeepAliveTimeout.

With KeepAlive set to true, command line SVN clients are very diligent in closing connections. However, SVNKit keeps connections open. As Apache documentation states, if KeepAlive is set to true, the client should use an existing connection, and close it when done. Giving a somewhat conflicting command, SVNKit opens a connection and keeps it open. On issuance of a subsequent command, SVNKit may open a new connection, regardless of how many established connections are still open.

To force a connection closure upon command completion, set the `KeepAliveTimeout` to a smaller value than the value for `Timeout`.

Generally, a `KeepAliveTimeout` value of 15 seconds works for WANdisco products. If your application ends up dropping and then establishing the connection because of this low value, you may have to increase the `KeepAliveTimeout` value. (For example, you may notice a larger number of pending transactions on the Admin Console's Dashboard page.) Under a normal load, a client follows a request within 15 seconds, but under a heavy load or no load, the number of seconds could vary widely, depending on your specific configuration.

The server sends the `Timeout` value to the client as a part of the response header, and the client uses this value to reset or resend the command if the server does not reply within that time. With a low value for `Timeout` (for example, 120 seconds), if the server for some reason does not complete the action and does not reply back, the client sends the command again on a different, newly established connection. If this happens with WANdisco, WANdisco ends up replicating an unnecessary transaction, and may not parse correctly with an invalid response.

WANdisco also requires some extra time for transaction coordination. In a singleton quorum, with a client connecting to the distinguished node, the transaction overhead is about 300 mille seconds. To be on the safe side, WANdisco recommends you set the `KeepAliveTimeout` value much higher than 300 mille seconds. With `KeepAlive` set to true, and an appropriate value for `KeepAliveTimeout`, any Apache worker threads and lingering connections should be taken care of.

**NOTE:**

---

WANdisco does not recommend setting `KeepAlive` to false. If you set `KeepAlive` to false, a client's transactions have an enormous overhead of establishing and destroying the connection. This overhead exists regardless of WANdisco.

---

## 9.3 Error Messages

### 9.3.1 Missing License Key File

Subversion Access Control depends on a license key file being present in the `svn-security/config` directory. Please get a valid license from WANdisco and copy the file to the config directory. Access Control does not start without the license file.



## 9.3.2 Client could not read status line: Server Closed Connection

You may encounter an error such as:

```
Apache Error: Client could not read status line: Connection was closed by
server
```

```
WANdisco Logs: AssertionError Activity is null for xxxx
```

The solution may be toggling an element. In the Apache configuration files, set `Keepalive` to `On`. If this does not remedy the situation, add the `ConnectionReset` element and set to `true` in the `SVNProxy` element of the `prefs.xml`. Please refer to [9.2, Apache and SVNKit](#).

## 10 Frequently Asked Questions

---

### 10.1 Why Are So Many Java Processes Running?

On older versions of Linux, every thread is listed as a process by the `ps` command. This does not affect the operation of Access Control. WANdisco does not support the older versions of Linux.

### 10.2 Can I Store Logs or Content on NFS?

NFS (Network File System) allows files and directories to be accessed remotely over a network using NFS clients. NFS clients are typically built into the operation system kernel these days. However, some operations, like renaming a file, are not guaranteed to be atomic over NFS. Here is a snippet from the `rename` function's `man` page on Linux, for example:

**BUGS**

```
On NFS filesystems, you can not assume that if the operation failed the file was not renamed. If the server does the rename operation and then crashes, the retransmitted RPC which will be processed when the server is up again causes a failure. The application is expected to deal with this. See link(2) for a similar problem.
```

Code management systems such as Subversion make heavy use of the `rename` operation to modify the underlying databases. Independent of WANdisco, it is a risky practice to store Subversion database content on NFS. The code management community at large recommends not using NFS for storing repositories.

Some WANdisco products are bundled with a built-in transactional journal and an object database. These are by default stored in the `cvs-security/systemdb` and `cvs-security/config` directories. These directories should not be mounted on an NFS drive. The replicator itself may be installed on an NFS drive but the `systemdb` and `config` directories should be on direct storage (non-NFS options like RAID, SCSI, SAN, etc). Replicator's transactional integrity can be compromised if writes to an NFS server are lost due to a potential NFS client cache crash after the NFS server has indicated IO completion.

### 10.3 Why is Set Up Configuring IP Addresses as 0.0.0.0?

The address 0.0.0.0 is a special IP address, treated as a wild-card IP address. In other words, on a machine with multiple NICs (Network Interface Cards), it binds to all interfaces. The advantages of using wild-card IP address include:

- It avoids binding to a fixed IP address. If the host's IP address changes, (for example, the subnet changes, or the machine is moved to a different location) you don't have to change the wild-card IP in the prefs.xml file to the new IP address.
- There is wider bandwidth to TCP clients. Now TCP clients can connect to any NIC, because WANdisco is listening on multiple NICs.

The disadvantage to using the wild-card IP is that it gives coarser access control at the IP address level, as all address are being listened to at the specified port.

You can always switch from the wildcard IP address to a fixed, static IP address or a DNS host-name, though for the most part, WANdisco recommends you stick with wild-card addressing.

## 10.4 WANdisco Authentication

Authentication is the process of determining whether someone or something is, in fact, who or what it is declared to be. Authorization is the process of giving someone permission to do or have something.

The Apache user-names and passwords should match at all sites. The WANdisco Subversion replicator's license manager requires a valid user-name inside the HTTP authorization header to be passed for all DAV commands, except `OPTIONS` and `PROPFIND`. In other words, anonymous access to Apache is not allowed to enforce license requirements, unless you have an unlimited or an evaluation license. With an unlimited or evaluation license, you are not required to register the user. This typically means ensuring a `Require valid-user` line is specified in the Apache SVN DAV configuration files in the `/etc/httpd/conf/httpd.conf` and `/etc/httpd/conf/conf.d/*` directories. When using Basic Authentication, it is the end user or administrator's responsibility to keep Apache authentication databases in sync across all sites.

## 10.5 Apache 2.2 with SVNDAV on Windows

- Step 1 Install Apache2.2.
- Step 2 Install svn-win32-1.4.4 for Apache 2.2. Make sure it's Subversion for Apache 2.2.
- Step 3 Copy `svn-win32-1.4.4/bin/intl3_svn.dll` to `apache/bin`.
- Step 4 Copy `svn-win32-1.4.4/bin/libdb44.dll` to `apache/bin`.
- Step 5 Copy `svn-win32-1.4.4/mod_authz_svn.so` to `apache/modules`.
- Step 6 Copy `svn-win32-1.4.4/mod_dav_svn.so` to `apache/modules`.

Step 7 Uncomment these lines in `apache/conf/httpd.conf`:

```
LoadModule dav_module modules/mod_dav.so
LoadModule dav_fs_module modules/mod_dav_fs.so
```

Step 8 Add these lines to `apache/conf/httpd.conf`:

```
LoadModule dav_svn_module modules/mod_dav_svn.so
LoadModule authz_svn_module modules/mod_authz_svn.so

<Location /myDavLocation>
DAV svn
SVNPath C:\repo
SVNAutoversioning on
AuthType Basic
AuthName "SVN Repo"
AuthUserFile C:\repo\dav-auth
Require valid-user
</Location>
```

Step 9 Check that the users have been added to the `C:\repo\dav-auth` file. To add new users or change passwords, use `apache/bin/htpasswd.exe`.

Step 10 Restart Apache.

Step 11 Point a web browser to: `http://server:port/myDavLocation`.

## 10.6 Setting Up Apache for SVN-DAV

Sometimes, it is required that Apache server be running on port 80 and server multiple locations in addition to SubVersion. This can be accomplished with proxy enabled.

This section outlines the steps to set up Apache for such configuration and also for setting up SVN-DAV for HTTPS access. The reader is assumed to be familiar with the Apache set up and basic WANdisco replicator set up. The assumption is that Apache is running on UNIX, though the same steps apply for the Windows platform. These points are assumed:

- Apache server is running on port 80
- Apache web-dav module is running on port 8181
- WANdisco Access Control is configured to listen on port 8080, and it forwards the requests to apache web-dav module on port 8181
- WANdisco is listening on port 6444
- Apache SSL is running on port 443 to handle HTTPS requests (if this is set up, the stunnel package is not required)
- All the processes are running on the same machine
- Apache is compiled with `mod-proxy` and `mod-ssl` modules
- The SVN URL is `/svnrepos` (either as a parent-path or path)

Run the `<svn-security>/bin/setup` utility and specify the following ports for each replicator:

- DConE port 6444
- Proxy port 8080
- Apache web-dav port 8181 on localhost

In the `httpd.conf` file, specify the following parameters:

```
#

# Define apache port and pass anything that matches location /svnrepos to
# WANDisco SVN Replicator

#

    NameVirtualHost *:80
<VirtualHost *:80>
    ProxyPass /svnrepos http://127.0.0.1:8080/svnrepos
    ProxyPass !svn http://127.0.0.1:8080/svnrepos!/svn
    ProxyPassReverse /svnrepos http://127.0.0.1:8080/svnrepos
    ProxyPassReverse !svn http://127.0.0.1:8080/svnrepos!/svn
    RequestHeader edit Destination ^https: http: early
</VirtualHost>

    Listen 443

<VirtualHost *:443>
    ProxyPass /svnrepos http://127.0.0.1:8080/svnrepos
    ProxyPass !svn http://127.0.0.1:8080/svnrepos!/svn
    ProxyPassReverse /svnrepos http://127.0.0.1:8080/svnrepos
    ProxyPassReverse !svn http://127.0.0.1:8080/svnrepos!/svn
    RequestHeader edit Destination ^https: http: early
    SSLEngine on
    SSLCertificateFile /etc/httpd/conf/ssl.crt/server.crt
    SSLCertificateKeyFile /etc/httpd/conf/ssl.key/server.key
    SSLCACertificateFile /etc/httpd/conf/ssl.crt/ca-bundle.crt
</VirtualHost>

    Listen 8181
        NameVirtualHost *:8181
<VirtualHost *:8181>
    KeepAlive On
<Location /svnrepos>
    AllowOverride None
    Order allow,deny
    Allow from 127.0.0.1
    DAV svn
    SVNParentPath /tmp/dav
    AuthType Basic
    AuthName wandisco
```

```
AuthUserFile /etc/httpd/conf/htpasswd  
Require valid-user
```

## 10.7 Encryption Around WANdisco Protocol

Here are the details about any ECCN classifications you may have applied for and been granted from the U.S. Government for export (due to the encryption capabilities in the client for DAV over SSL).

WANdisco Subversion Access Control distribution does not actually perform any encryption or decryption of the DAV traffic. We rely on Apache to decrypt the SSL traffic, and then use a proxy-pass definition within the Apache configuration to redirect the unencrypted request to the WANdisco replicator.

The WANdisco replicators do not directly encrypt communication between sites. Instead, many customers may use something like a persistent VPN connection for the replicator-to-replicator traffic over an encrypted connection, but our code actually does no encryption.

Lastly, the WANdisco replicator simply sits as a proxy on the Subversion server itself (the host running Apache and ModDAV), so there is no client component that we provide that would be sending any traffic to the Subversion server.

## 10.8 How Do I Restrict Direct Access to My Repository?

If you would like to prevent users from directly accessing your Subversion repository, use the Subversion `Location` directive as suggested below. You allow only specific IP addresses to access the repository.

This assumes that WANdisco and Apache server are running on the same machine.

From the example shown in [10.5, Apache 2.2 with SVNDAV on Windows](#):

```
<Location /svnrepos>  
AllowOverride None  
Order allow,deny  
Allow from 127.0.0.1  
DAV svn  
SVNParentPath /tmp/dav  
AuthType Basic  
AuthName wandisco  
AuthUserFile /etc/httpd/conf/htpasswd  
Require valid-user  
</Location>
```

## 10.9 About WANdisco Log Files

WANdisco uses Java logging. See <http://java.sun.com/j2se/1.4.2/docs/guide/util/logging/overview.html> for a discussion on Java logging. Make any changes to the `svn-replicator/config/log.properties` file.

WANdisco places the log files in `svn-replicator/logs`. The current file is always `SVNProxy-Server-prefs.log.0`, and the files are rotated out and eventually garbage collected. For rotation schedule, see the `svn-replicator/config/log.properties` file.

Here is a brief explanation of Java logging. The newest log is always `log.0`. When that log reaches a specified size (500 KB by default), that log gets renamed to `log.1` and a new `log.0` is started. The old `log.1` becomes `log.2`, `log.2` becomes `log.3` and so on. The second newest log is always `log.1`. WANdisco's Log Viewer displays a drop-down list showing the other logs. As the log file names increment higher, they represent going further back in time.

If you want to change the defaults on the file size before log rotation and how many logs to keep, change these parameters in `svn-replicator/config/log.properties`:

```
java.util.logging.FileHandler.limit = 500000
java.util.logging.FileHandler.count = 500
```

The limit is in bytes and the count is the maximum number of logs to keep. Any changes to `log.properties` are unique to each site, and are not replicated.

# Appendix A - Installing Java and Perl

---

You should have already installed Java and Perl at all the sites in your replication group for your trial evaluation. However, any new site you add to the replication group needs Java and Perl installed as well.

## Installing Java

- Step 1 Install JDK 1.5 and define the `JAVA_HOME` environment variable to point to the directory where the JDK is installed. You can download JDK 1.5 from the URL below.

```
http://java.sun.com/javase/downloads/index_jdk5.jsp
```

- Step 2 Add `$JAVA_HOME/bin` to the path and ensure that no other java (JDK or JRE) is on the path.

```
$ which java
/usr/bin/java
```

```
$export JAVA_HOME="/usr"
```

or

```
$which java
/export/share/apps/jdk/1.5.0/bin/java
```

```
$export JAVA_HOME="/export/share/apps/jdk/1.5.0"
```

- Step 3 Ensure the full JDK is installed, not just the JRE. This can be confirmed by running `java -server-version`. If it generates a **not found** error, repeat Steps 1 and 2.

If you find package management problems or conflicts with the JDK version you are downloading (for example, rpm download for Linux), you may want to use the self-extracting download file instead of the rpm (on Linux) package. The self-extracting download easily installs in any directory without any dependency checks.



## Installing Perl

- Step 1 On UNIX or Cygwin, install perl version 5.6 or greater and ensure that the perl executable is on the system path.
- Step 2 On Windows, install ActivePerl version 5.8 or greater and ensure that the perl executable is on the system path. You can download the MSI installer for ActivePerl from the URL below.

<http://activestate.com/Products/Download/Download.plex?id=ActivePerl>.