

TACACS.net™

Installation & Configuration Guide

Table of Contents

1. Introduction.....	2
2. System Requirements	2
2.1. Base System Requirements	2
2.2. For Best Results.....	2
3. Installation	2
3.1. Installation Wizard	3
3.2. Confirming Your Installation	3
3.3. Starting and Stopping Services	3
4. Configuration	4
4.1. Global Configuration.....	4
4.1.1. Port	4
4.1.2. LocalIP	4
4.1.3. Logging	4
4.1.4. Syslog	5
4.2. Authentication	5
4.2.1. Configuring Authentication Using Local Service (File Group) Users.....	5
4.2.2. Configuring Authentication Using Localhost Users.....	6
4.2.3. Configuring Authentication Using Active Directory.....	6
4.2.4. Configuring Authentication Using Active Directory and LDAPS	7
4.2.5. Configuring Authentication Using LDAP	8
4.2.6. Configuring Authentication using RADIUS.....	9
4.2.7. The DEFAULT User Group.....	10
4.2.8. Creating Encrypted Passwords.....	10
4.3. Clients	11
4.3.1. Default Client Groups	11
4.3.2. Configuring your router or switch.	12
4.4. Authorization.....	12
4.4.1. Time	12
4.4.2. User Groups.....	13
4.4.3. Client Groups	13
4.4.4. AutoExec	13
4.4.5. Shell.....	13
4.4.6. Services.....	13
4.4.7. RemAddr.....	13
4.4.8. The Local System Administrators Profile	13
4.4.9. The DEFAULT Authorization Profile	14
5. Testing the Server	14
5.1. TACVerify.....	14
5.2. TACTest.....	15
5.2.1. TACTest Examples	17
6. Windows Firewall.....	17
7. Optimizing Performance	18
8. Feature Enhancements.....	19
9. Recommended Tools.....	19

1. Introduction

Thank you for choosing the TACACS.net TACACS+ Server! TACACS.net is the simplest, easiest, most flexible, and most cost efficient TACACS+ server for Windows PCs and Servers. This software was designed by network administrators for network administrators and can be used in SOHO, SMB, Enterprise, WAN, or lab environments for setting granular access policies to network devices. For more information and documentation, please visit our web site at www.TACACS.net.

2. System Requirements

2.1. Base System Requirements

- Windows XP, Windows 2000 Workstation or Server or later. ¹
- 1 GHz CPU
- 256 MB RAM
- 500 MB HDD free

2.2. For Best Results

- The software will run on Windows XP and Windows 2000, but runs best on Windows 7 or Windows 2008 Server or later.
- If you plan to use Active Directory authentication, install on a Windows Server that is configured as a Domain Controller, Read-Only Domain Controller, or Member Server.
- Use the default installation directories.
- Enable DNS services on the same server and add the network devices (TACACS+ clients), or configure it to do a zone transfer from name servers with this information. This will enable authorization by host names and speed up requests.
- Do not install any other software on the server that TACACS.net is installed on. It is not possible to test compatibility with every other software application, so there is no way to tell for sure that 3rd party software won't conflict with the server or cause unexpected results.
- Do not enable any other services that are not required on this server. Another program could conflict with the software, make it more difficult to troubleshoot, or slow down the server. Also, a vulnerable service could expose the TACACS+ Server to unnecessary security threats.
- Use the Windows firewall to block any unused ports.
- Go through all the configuration instructions and test with the included TACTest client before testing with any network devices.

3. Installation

TACACS.net was designed from the bottom up to be easy to use and configure. In most cases, you should be up and running within 10 minutes!

1. Download the software from www.tacacs.net.
2. Extract the installer from the .zip file.
3. Optional but recommended: Run MD5 sum to confirm the software is correct and hasn't become corrupted while downloaded. There are many free tools available on the Internet to check MD5 file hashes.
4. Run the installation Wizard.

¹ TACACS.net 2.x, also known as TACACS.net Advanced, requires Windows Vista or Windows Server 2008 or later.

3.1. *Installation Wizard*

An installation wizard is provided to install TACACS.net. The installation process will do the following:

1. Install program files in their default locations.
2. Register the Path for scripts and binaries.
3. Install Microsoft .NET if needed.
4. Create Start menu items.
5. You will also have the option of setting the shared secret for your deployment. This can be changed later if you like in clients.xml.
6. Follow the easy step-by-step prompts while installing the software.
7. Read and accept the End User License Agreement.
8. Review the Readme file for general information about your installation including installation locations and default settings.

3.2. *Confirming Your Installation*

After installing the TACACS.net TACACS+ server, it will start by default. You can confirm its installation in a couple of places:

1. Start > Control Panel > Administrative Tools > Services.
2. Using the context menu on the taskbar and selecting "Task Manager" or using the key combination Ctrl+Shift+Esc. You will find the executable 'tacplus.exe' under the Processes tab.
3. Running Netstat from the command line.

```
C:\>netstat -ab
Active Connections
Proto Local Address      Foreign Address    State             PID
...
TCP    mypc:49            mypc:0            LISTENING        2860
[tacplus.exe]
```

Figure 1: TACACS.net in Netstat

3.3. *Starting and Stopping Services*

You do not need to restart the service after making a configuration change². The server will automatically re-read the configuration files when they are edited.

1. You can start and stop services from the Services Management Console by going to: Start > Control Panel > Administrative Tools > Services.
2. You can also start and stop services from the command line by using the net stop/net start commands.

```
C:\>net stop tacacs.net
The TACACS.net service is stopping.
The TACACS.net service was stopped successfully.

C:\>net start tacacs.net
The TACACS.net service is starting.
The TACACS.net service was started successfully.
```

Figure 2: Starting & stopping TACACS.net from the command line.

² Restart required when making changes to the global configuration file tacplus.xml.



Read the Quickstart Guide to get your server up and running and confirm basic operation, and then return to this guide for further information on configuring and managing your server.

4. Configuration

The configuration files should now be accessible from the Programs menu at Start > All Programs > TACACS.net > Configuration. These files are in XML format and simple to modify with any text editor like Notepad or Wordpad or an XML editor. You will find instructions in the configuration files themselves in addition to the instructions in this guide. All files are read by the software linearly (from top to bottom), so if there is a conflict, the first entry will take precedence.



Before changing any of the default configuration files, make a backup of the originals so you can restore them later if needed. Copy the original files to a directory named YYMMDD_vvv_orig (eg; 100115_132_orig) in case you need them again in the future. Whenever you make a change to a file already in production, rename the current version to NAME.xml.YYMMDD (eg; authentication.xml.110504) so that you can restore that version if needed.

4.1. Global Configuration

The global configuration for TACACS.net is in tacplus.xml. Most deployments will not need to make any changes to this file, but there are some elements that you should be aware of:

4.1.1. Port

The TCP port that the server uses is defined in <Port>. The TACACS+ protocol specification defines TCP port 49 for use for TACACS+, and it is recommended to keep this port. Many TACACS+ network device clients cannot use other ports, so changing this could introduce unnecessary troubleshooting problems.

4.1.2. LocalIP

This is the IP address that the Server will use. By default, this is set to 127.0.0.1. You should change this to the server's IP address if you have multiple physical or virtual interfaces or IP addresses, if you have installed the software in a virtual machine like VMWare, or if you get socket errors when running TACTest.

4.1.3. Logging

These settings define the name, location, logging level, and rollover settings for the logs. The following logging levels are available: Alert, Critical, Error, Warning, Notice, Information, and Debug. Debug generates the most information, and Alert generates the least amount of logging information. RolloverDays specifies how many days to keep logs before starting a new log. RolloverMB specifies the maximum size the log file can get before rolling over, and DeleteDays specifies how many days to keep files before automatically deleting them.³

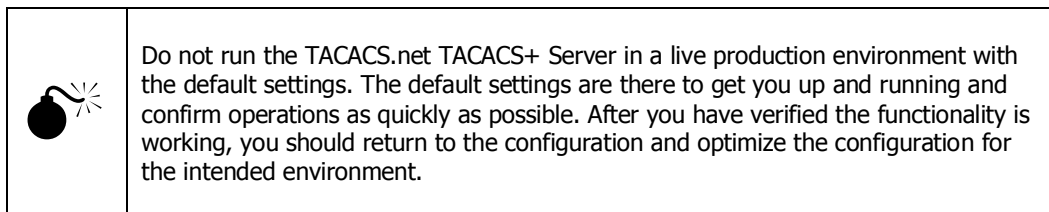
³ Enhanced logging was introduced in v 2.1.0. Refer to the Enhanced Logging Configuration Guide for information on configuring Enhanced Logging.

4.1.4. Syslog

This setting is used if you would like to log to an external Syslog server. Syslog support was added in version 1.2. If you have a previous version of the tacplus.xml configuration file, you can download the updated configuration file from the web site or add this line manually. These settings were deprecated in version 2.x to enable the enhanced logging functionality with structured logging and granular control over syslog destinations, severity levels, and content. Syslog settings for 2.x are configured in logging.xml.

4.2. Authentication

All authentication settings can be found in authentication.xml.



4.2.1. Configuring Authentication Using Local Service (File Group) Users

Users may be configured within the TACACS.net server and can work simultaneously with other user groups such as Active Directory users. These users are useful for role accounts, scripted tools, or other users that are not part of another authentication database. Because they are the simplest to set up and have no dependencies outside of TACACS.net, it is recommended to configure these users first to confirm your TACACS.net installation.

Near the top of authentication.xml, you will find a File Group example commented out. Simply remove the comments in the UserGroup section (highlighted in yellow below) and that will activate the two example users (user1 and user2). After you have done that, run TACTest to verify server operation, and then modify those two users to create your local File Group users.

```

<UserGroup>
  <Name>Network Engineering</Name>
  <AuthenticationType>File</AuthenticationType>
  <!--
    <Users>
      <User>
        <Name>user1</Name>
        <LoginPassword ClearText="somepassword" DES=""> </LoginPassword>
        <EnablePassword ClearText="" DES=""></EnablePassword>
        <CHAPPassword ClearText="" DES=""> </CHAPPassword>
        <OutboundPassword ClearText="" DES=""> </OutboundPassword>
      </User>
      <User>
        <Name>user2</Name>
        <LoginPassword ClearText="somepassword" DES=""> </LoginPassword>
        <EnablePassword ClearText="" DES=""></EnablePassword>
        <CHAPPassword ClearText="" DES=""> </CHAPPassword>
        <OutboundPassword ClearText="" DES=""> </OutboundPassword>
      </User>
    </Users>
  -->
</UserGroup>

```

Figure 3: Use the File Group Example to create your own File Group users.

4.2.2. Configuring Authentication Using Localhost Users

Localhost users are users configured within Windows Users and Groups on the local computer. These are the user accounts that are authorized to log in to this local computer from the standard Windows login screen. This does not apply to users configured within Active Directory on a Domain Controller. This is a common scenario if TACACS+ is going to run from one computer in a lab environment or small office/home office (SOHO) scenario. Localhost User Groups are easy to configure. The example provided (Local System Administrators) is enabled by default to permit testing and verification. Make modifications to this example to match the needs of your installation. For example, modify the <Name> to match the name you would like to use for this group. Modify the <LocalhostGroupName> to match the name of the User Group in Windows Users and Groups that you would like to use for authentication.

```
<UserGroup>
  <Name>Local System Administrators</Name>
  <AuthenticationType>Localhost</AuthenticationType>
  <LocalhostGroupName>Administrators</LocalhostGroupName>
</UserGroup>
```

Figure 4: Localhost Users

See also: the DEFAULT User Group below.

4.2.3. Configuring Authentication Using Active Directory

Windows Active Directory groups will authenticate using a Windows Domain Controller on the same server or on a different server. For best results, run TACACS+ on a member server in the domain.



NOTE: It is strongly recommended to install TACACS+ on the server that holds the user database. If there is a connection problem between the TACACS+ service and another server that serves the user database, the TACACS+ server will not find the user and will send back a reject to the Client. Depending on how the Client is configured, if the Client receives a denied authentication response to a TACACS+ request, it will not fail over to local authentication, preventing all logins except for those using a different database (such as local configuration file). This is especially relevant when using Active Directory because Active Directory uses Kerberos, which depends on closely synchronized time between the TACACS+ server and the Domain Controller. This is true for any TACACS+ server, and is not specific to TACACS.net.

To see the user directory subtree information, you can execute 'dsquery' from the command line on Windows Server:

```
C:\>dsquery user -samid USERNAME
```

To see the list of AD groups the user belongs to, use:

```
C:\>dsquery user -samid USERNAME | dsget user -memberof -expand
```



The 'dsquery' command is only available on Windows Server.

The User Group <Name> is used to match a policy in authorization.xml. To avoid confusion, use the same name as the Security Group name in Active Directory.

The <LDAPUserDirectorySubtree> Enter the distinguished name (DN) of the user directory subtree that contains all users. Copy and paste the output of 'dsquery' for the configuration parameters without using the CN=USERNAME.

The <LDAPGroupName> will come from the output of the 'dsquery' command. You can use the complete DN of the group or just the AD name of the group in the <LDAPGroupName> configuration parameter.

<LDAPAccessUserName> and <LDAPAccessUserPassword> are optional elements and should be specified if the Active Directory server does not allow anonymous access to the Active Directory for authentication. This username must have read/write access to Active Directory.

The following example shows the results of using the 'dsquery' command for user 'steve' on server 'myserver' in domain 'lab.contoso.com'.

```
C:\>dsquery user -samid steve
"CN=steve,CN=Users,DC=myserver,DC=lab,DC=contoso,DC=com"

C:\>dsquery user -samid steve | dsget user -memberof -expand
"CN=Support,CN=Users,DC=myserver,DC=lab,DC=contoso,DC=com"
"CN=Domain Users,CN=Users,DC=myserver,DC=lab,DC=contoso,DC=com"
"CN=Users,CN=Builtin,DC=myserver,DC=lab,DC=contoso,DC=com"
```

Figure 5: Output of 'dsquery'

The following example demonstrates an Active Directory authentication group using the output of 'dsquery' above.

```
<UserGroup>
  <Name>Support</Name>
  <AuthenticationType>Windows_Domain</AuthenticationType>
  <LDAPServer>127.0.0.1:389</LDAPServer>
  <LDAPUserDirectorySubtree>CN=Users,DC=myserver,DC=lab,DC=contoso,DC=com</LDAPUserDirectorySubtree>
  <LDAPGroupName>Support</LDAPGroupName>
  <LDAPAccessUserName>Administrator</LDAPAccessUserName>
  <LDAPAccessUserPassword ClearText="mypassword" DES=""></LDAPAccessUserPassword>
</UserGroup>
```

Figure 6: AD configuration example

The server can only authenticate a user to the group that user is directly a member of. It cannot authenticate a user to a group that is higher or lower in the AD tree. For example, if your user is in dc1.contoso.com/Users/Group1/SubgroupA, you must reference the group 'SubgroupA' in your configuration. The user will not authenticate using the group 'Group1'. This enables the administrator to set granular access policies and makes the server run faster.

4.2.4. Configuring Authentication Using Active Directory and LDAPS

LDAPS (LDAP over SSL/TLS) can be used to add additional security to sessions. LDAPS is the default for Windows Server 2008 and later. Note the <LDAPUseSSL> option in the configuration below.

```

<UserGroup>
  <Name>group_w</Name>
  <AuthenticationType>Windows_Domain</AuthenticationType>
  <LDAPServer>ad.mydomain.net:636</LDAPServer>
  <LDAPUseSSL>1</LDAPUseSSL>
  <LDAPUserDirectorySubtree>cn=OurUsers, DC=mydomain, DC=net
</LDAPUserDirectorySubtree>
  <LDAPGroupName>CN=admin, OU=ROUTERS, OU=TACACS, OU=ManagedServices,
  DC=mydomain, DC=net</LDAPGroupName>
  <LDAPAccessUserName>someBindUser</LDAPAccessUserName>
  <LDAPAccessUserPassword ClearText="mypassword" DES="">
</LDAPAccessUserPassword>
</UserGroup>

```

Figure 7: Sample LDAPS Configuration

Notes for the configuration above:

1. The FQDN of the AD server must be resolvable by the client and must be part of the dns= field of the SubjectAlternativeName in the X.509 generated certificate. In other words, you have to test the connection to port 636 using the hostname with some type of LDAPS client (eg Softerra's LDAP Browser). The trust path for the certificate also needs to be in place, as the .NET library validates the CAs in the trust chain before making a connection.
2. You can't use the IP address in the LDAPServer field here unless that IP address is also in the dns= field of the SubjectAlternativeName field (also called the SubjectAltName in RFC 6125).

4.2.5. Configuring Authentication Using LDAP

Configuring authentication using LDAP is similar to the configuration for using Active Directory. Instead of using "Windows_Domain" for <AuthenticationType>, you will use "LDAP". There are also 4 additional elements to configure:

1. <LDAPUserObjectClass>
2. <LDAPUserNameAttribute>
3. <LDAPMemberOfAttribute>
4. <LDAPAuthType>

Refer to the documentation for your LDAP server to find this information for your deployment.

The following is an example configuration using the iPlanet LDAP server.

```

<UserGroup>
  <Name>group_x</Name>
  <AuthenticationType>LDAP</AuthenticationType>
  <LDAPAuthType>Basic</LDAPAuthType>
  <LDAPServer>x.x.x.x:389</LDAPServer>
  <LDAPUserDirectorySubtree>o=company, dn=etc.</LDAPUserDirectorySubtree>
  <LDAPMemberOfAttribute>groupMembership</LDAPMemberOfAttribute>
  <LDAPGroupName>cn=group_x, ou=etc, dn=etc</LDAPGroupName>
  <LDAPUserNameAttribute>uid</LDAPUserNameAttribute>
  <LDAPAccessUserName>cn=admin, o=etc, dn=etc.</LDAPAccessUserName>
  <LDAPAccessUserPassword ClearText="password" DES=""></LDAPAccessUserPassword>
</UserGroup>

```

Figure 8: Sample iPlanet LDAP server configuration

Notes for the iPlanet configuration above:

1. `<LDAPAuthType>Basic</LDAPAuthType>`
This is required to change the connection type, which is NTLM by default.
2. `<LDAPMemberOfAttribute>groupMembership</LDAPMemberOfAttribute>`
This is the name of the attribute used to define profiles in iPlanet.
3. `<LDAPGroupName>cn=group_x, ...</LDAPGroupName>`
To check the contents of "groupMembership", for each group_x defined at TACACS.net.

4.2.6. Configuring Authentication using RADIUS

RADIUS was added in TACACS.net 2.3. It can be used to migrate from RADIUS to TACACS+, authenticating to a customer or partner's users, or using a 3rd party Multi-Factor Authentication system. Multiple failover RADIUS servers can be configured for a UserGroup, but they should have the same settings. Following are some of the configuration options.

1. If StripRealm is Enabled, it will remove the realm suffix before sending the username to the RADIUS server.
2. TimeoutSecs is how long TACACS.net will wait for a response before it times out and tries again.
3. ConnectionAttempts is the number of attempts it should make before moving on to the next server or UserGroup.
4. Authenticate any user on the RADIUS server or specify which users should be authenticated by matching on a RADIUS Attribute/Value pair. This is an optional configuration setting. TACACS.net supports the standard IETF attributes defined in RFC 2865 Ch.5. If you would like to specify only particular users to be authenticated from a particular RADIUS server, add the Reply-Message and configure a string to look for eg; "TACACS" or "TACACS-NOC", etc. You can set multiple attributes and the authenticated users would have to match all attribute/value pairs. The AttributeMatch is case sensitive unless you use (?i) at the beginning of the string.
5. You can set realms to be used for this UserGroup. This is an optional setting. This UserGroup will not be used unless the specified realm(s) are sent as a part of the username. If a realm is set, it must be used by the user in order to authenticate.

```
<UserGroup>
  <Name>RADIUS Users</Name>
  <AuthenticationType>RADIUS</AuthenticationType>
  <RADIUSStripRealm>Disabled</RADIUSStripRealm>
  <RADIUSServer>192.168.1.1:1812</RADIUSServer>
  <RADIUSServer>192.168.1.2:1812</RADIUSServer>
  <RADIUSSharedSecret ClearText="mysecret" DES=""></RADIUSSharedSecret>
  <RADIUSTimeoutSecs>1</RADIUSTimeoutSecs>
  <RADIUSConnectionAttempts>1</RADIUSConnectionAttempts>
  <RADIUSAttributeMatch>(?i)Reply-Message=.*TACACS.*</RADIUSAttributeMatch>
  <RADIUSRealm>.*@realm1.com.*</RADIUSRealm>
  <RADIUSRealm>.*@realm2.*</RADIUSRealm>
</UserGroup>
```

Figure 9: Sample RADIUS Configuration

4.2.7. The DEFAULT User Group

The DEFAULT group is included with the installation. It is the fallback group that will allow all users configured in the Administrators group on the local computer as TACACS+ users by default. This group is added so that an administrator installing TACACS+ on a standalone server can be up and running immediately with minimal configuration. This group should be commented out if not needed in a production environment.

4.2.8. Creating Encrypted Passwords

When first testing the server and verifying operations, you should use ClearText passwords in the above configuration files, but once you move to production, you should use encrypted passwords instead. If both a ClearText and a DES password is specified in the configuration, the ClearText password will take precedence. TACACS.net includes a password encryption utility called TACDES to create encrypted passwords.

4.2.8.1. TACDES

The TACDES tool is simple to use. Simply launch TACDES from the Start Menu item in the TACACS.net directory at Start > All Programs > TACACS.net > TACDES and type tacdes and the password you want to encrypt. Then copy and paste the new password into your configuration.

```
TACDES 1.0 (C) TACACS.net
Type tacdes -? for help.

C:\Program Files\TACACS.net>tacdes -?
TACDES 1.0 (C) TACACS.net
A tool for generating DES encrypted passwords that can be used with TACACS.net TACACS+
server.

Usage: tacdes [list of passwords]

e.g., tacdes password1 -This encrypts the password 'password1'
e.g., tacdes password1 password2 -This encrypts the passwords 'password1' and
'password2'

C:\Program Files\TACACS.net>tacdes mypassword
Encrypted mypassword is Vm/Hiyhd8wHFXYJfDtR7w==

C:\Program Files\TACACS.net>exit
```

Figure 10: Using TACDES



TACDES is designed specifically for TACACS.net. Other tools will not work to create DES encrypted passwords for TACACS.net and TACACS.net TACDES will not create encrypted passwords for other software.

4.3. Clients

A TACACS+ client is a router, switch, firewall, or other network device that will send authentication requests to the TACACS+ server. Clients are also sometimes called a NAS (Network Access Server). In order for a client to work with TACACS+, the TACACS+ server needs to know that a specified client is authorized to send requests. These settings are configured in clients.xml. Clients may also be used with authorization configuration to specify policies per client or client type.

This file is read linearly (top to bottom). This means that the first match is applied. This will enable you to configure overlapping Device Groups. For example, you could specify one policy for 192.168.1.1/32 and another policy for 192.168.1.0/24 and the first match will be applied.

This configuration file supports Regular expressions. This gives the administrator additional flexibility in configuring clients. Regular expressions can be useful when you want to set policy based on hostnames instead of IP Addresses.

Here are some examples:

```
192.168.1.1..... This matches a single IP address.
192.168.* ..... This will match all ip addresses beginning with 192.168.
192.168.1.1-192.168.1.255 ..... This will match all ip addresses in the specified range.
192.168.0.0/16..... This will match all ip addresses in the specified CIDR format.
192.168.1.0/255.255.255.0 ..... This will match all ip addresses provided IP-Subnet configuration.
^OrgSwitch-a.* ..... This will match all hostnames with with the prefix 'OrgSwitch-a'.
^switch1$ ..... This matches a hostname 'switch1'.
switch..... This matches any hostname which has the word 'switch' in it.
```

For more information on using Regular Expressions refer to the following links:

- <http://www.regular-expressions.info/tutorialcnt.html>
- <http://www.regular-expressions.info/examples.html>
- <http://www.regextester.com/>
- <https://www.regex101.com/>

You will find more examples in the clients.xml file. Copy and paste one or more of the examples and make the necessary modifications to fit your needs.

For best results, put your more specific Clients first, and the less specific Clients towards the bottom of this file. Bear in mind that hostnames will require DNS to be available to the server running TACACS+. This may impact performance slightly, so don't use it if it's not necessary. For fault tolerance and performance, DNS should be running on the same server.

4.3.1. Default Client Groups

Several Default Client Groups are preconfigured to help you get up and running quickly. Once you move the server to a production environment, you should comment out these groups if they are not needed.

4.3.1.1. The LOCALHOST Client Group

The LOCALHOST Client Group is added by default to include the loopback address (127.0.0.1). This is one of the most commonly forgotten items when deploying a TACACS+ server, so we included it by default with the installation. It is important to have the loopback address included so you can run TACTest from this server.

4.3.1.2. The INTERNAL Client Group

The INTERNAL Group is added by default. This group will enable non-routable (RFC 1918) IP addresses to be TACACS+ clients without having to explicitly define them. This is useful in an internal NAT or lab network.

4.3.1.3. The DEFAULT Client Group

The DEFAULT group is added by default and is used as a catch-all group to verify operations of the server. This group will permit ANY network element to use TACACS+. This group should be removed or commented out before deploying the server in a production environment.

4.3.2. Configuring your router or switch.

Device manufacturers have implemented TACACS+ in different ways, so it is not possible to go into how to configure every type of router or switch for TACACS+. Refer to the documentation for your network device for information on configuring the device for TACACS+.

4.3.2.1. Sample Cisco router configuration

Following is an example configuration for Cisco IOS 12.2.

```
aaa new-model
aaa authentication login DEFAULT group tacacs+ line
aaa authorization console
aaa authorization config-commands
aaa authorization exec DEFAULT group tacacs+ none
aaa authorization commands 0 DEFAULT group tacacs+ none
aaa authorization commands 1 DEFAULT group tacacs+ none
aaa authorization commands 15 DEFAULT group tacacs+ none
aaa accounting commands 0 default start-stop group tacacs+
aaa accounting commands 1 default start-stop group tacacs+
aaa accounting commands 15 default start-stop group tacacs+
tacacs-server host <CLOSEST SERVER IP ADDRESS>
tacacs-server host <NEXT CLOSEST IP ADDRESS>
tacacs-server host <NEXT CLOSEST IP ADDRESS>
tacacs-server host <NEXT CLOSEST IP ADDRESS>
tacacs-server key <SHARED SECRET>
```

Figure 11: Sample Cisco IOS TACACS+ configuration

4.4. Authorization

Authorization is the functionality where you get to define policy based on the User, the Client, or time of day. Authorization policy allows you to specify which users have access to which devices and what commands they can run and when. Authorization is configured in authorization.xml. This file is read linearly. You can have multiple overlapping policies for the same Users or Clients. The first policy match will be the one applied. If the authorization.xml file has been renamed or deleted, all commands will be authorized. This can be useful for troubleshooting or in environments where there are only a few administrators and they all have the maximum privilege. An authorization policy includes the following elements:

4.4.1. Time

This is an optional element that is used to define a time period which this policy is in effect. You could have multiple policies for the same User Group so that during a specified time period (like a maintenance window) they have read/write privileges, but they have read-only privileges during the rest of the time. You would put the policy for the maintenance window first, and then the policy for all other times after

that. If no time settings are configured, the policy will always be in effect. The Days that can be used are: M(Monday), T(Tuesday), W(Wednesday), R(Thursday), F(Friday), S(Saturday), and N(Sunday). Time settings are based on military time, for example 07:00 is 7am and 17:00 is 5pm. The time used is the local time of the server TACACS+ is installed on.

4.4.2. User Groups

This User Group name must match a User Group name in authentication.xml. This tells the server to implement this policy when users in this User Group authenticate. A policy can be applied to multiple User Groups.

4.4.3. Client Groups


A Client Group is a group of network devices configured in clients.xml. If used, these Client Group names must match a Client Group name in clients.xml. A policy can be applied to multiple Client Groups. If no Client Group is specified, this policy will be applied to all Client Groups. You will see in the example that the Client Groups LOCALHOST and DEFAULT are used.

4.4.4. AutoExec

AutoExec is the settings you use when the user first connects to the client. This is run once when the user first logs in. This is where you could set a privilege level or an ACL to be applied or a command to be executed.

4.4.5. Shell

This is the section you define which commands are permitted and denied for this policy. Unlike the AutoExec section, the shell section is used continually during the session as the user is logged in. Regular Expressions are supported in shell commands. When the authorization.xml file is used, the default method is deny unless there is a permit rule. If you would like to end your ruleset with a permit all, use `<Permit>.*</Permit>`. If you would like to end your ruleset with a deny all, use `<Deny>.*</Deny>`.

	<p>The following commands will always be authorized: login, exit, quit, and end.</p>
---	--

4.4.6. Services

The Services section is used when someone is using TACACS+ to access a service or protocol on a client. This is also where you can define Vendor Specific Attributes. The services available are: slip, ppp, arap, tty-daemon, connection, system and firewall. The protocols available are: lcp, ip, ipx, atalk, vines, lat, xremote, tn3270, telnet, rlogin, pad, vpdn, ftp, http, deccp, osicp and unknown.

4.4.7. RemAddr

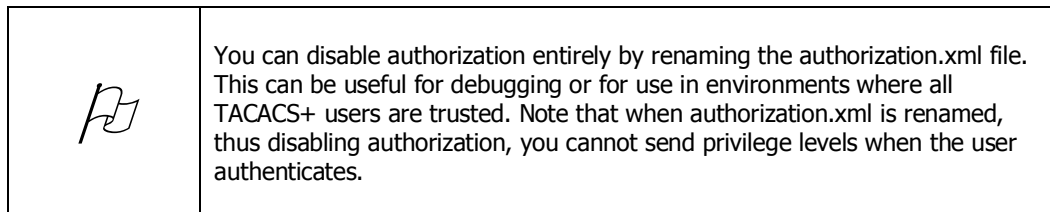
RemAddr was added in version 2.0.1. This allows the Administrator to set authorization profiles based on where the end user is coming from. A user can have separate privileges if they are connecting from an untrusted network or if they are connecting from within a trusted network. Privileged accounts can be restricted to specified IP addresses or subnets.

4.4.8. The Local System Administrators Profile

Take a look at the Local System Administrators profile near the bottom of this page. You will notice that the `<UserGroup>` Name (Local System Administrators) matches the name of an authentication group in authentication.xml. This means that when someone logs in with the Local System Administrators authentication group, the Local System Administrators authorization policy will be applied. This policy says that these users are permitted to use any show commands, but are denied any other commands.

4.4.9. The DEFAULT Authorization Profile

The DEFAULT Authorization Profile can be found at the bottom of this page. Unless this policy is commented out, this will allow all authenticated users to run 'show' commands, while denying any other commands.



5. Testing the Server

5.1. TACVerify

TACVerify is a tool used to verify the syntax of your configuration files. You should run TACVerify each time you change the configuration. It is very simple to use. Go to Start > All Programs > TACACS.net > TACVerify. TACVerify will run through the configuration files quickly looking for syntax errors. If it doesn't find any, it will output "No errors were found in the configuration."

Here is an example of TACVerify with a syntax error in authentication.xml line 42, character 57.

```
Reading: C:\Documents and Settings\All Users\Application Data\TACACS.net\Config\tacplus.xml
-----
Reading: C:\Documents and Settings\All Users\Application Data\TACACS.net\Config\authentication.xml

There is an error in XML document (42, 57).

Error details:Instance validation error: 'File blah' is not a valid value for AuthenticationDatabaseType.

Errors were found in configuration files. Please fix these errors and try again.
```

Figure 12: TACVerify

Press [Enter] to exit TACVerify.

5.2. **TACTest**

TACTest is a TACACS+ client that you can use to test TACACS+ requests and responses and for performance testing. Before you attempt to run TACACS+ on any external Clients in a lab or production environment, you must first run TACTest to verify that the system is working correctly. If TACTest fails, your external Clients will fail also.

TACTest is not specific to TACACS.net. It will work with any server that runs the TACACS+ protocol. It can also be installed and run as an independent program, without the TACACS.net server if desired. You can run TACTest from Start > All Programs > TACACS.net or simply from the command line.

To view information about TACTest, type 'tactest' at the command line with no arguments.

```
C:\>tactest
TACTest 1.0.4143.32116 (C) TACACS.net
Type tactest -? for help.
```

To print out a list of command options and examples, type '**tactest -?**'.

```

C:\>tactest -?
TACTest 1.0.4143.32116 (C) TACACS.net
A tool for testing TACACS+ server responses.
This host must be in the server's authorized client list to work.

Usage: tactest [options]

Options:
-\?      Display help
-s       ServerIP IP      (If this is not provided then 127.0.0.1 is used)
-port    ServerIP Port   (If this is not provided then port 49 is used)
-k       Shared Key      (If this is not provided then no encryption is used)
-u       Username
-p       Password
-np     New Password     (used only for change password commands)
-type    Authentication type. Can be ASCII or PAP, CHAP Default is ASCII
-en      This sends an enable command to the server
-c       Send this many requests. Default is 1
-m       Send repeatedly for this many seconds.
-t       Send this many requests per second.
-r       Retries
-w       Wait time between retries in seconds.
-f       Input file to be used.
-service This is used to request authorization AV pairs from server
-command This is used to request authorization of a command from server
-authen  This is used to send authentication commands to the server. This is
the default command.
-acct    The type of accounting command to send. Valid values are start, stop &
watchdog
-author  This is used to send authorization commands to server or to request
authorization AV pairs from the server

Input file can be used for commands e.g., tactest -f filename.txt
If input file is used then the 't' option must be specified at command line
e.g, tactest -f filename.txt -t 20

Authentication Examples:
tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword
tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword -c 20
tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword -t 20
tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword -m 5
tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword -m 5 -t 20

Accounting Examples:
tactest -s 127.0.0.1 -k mykey -u myuser -acct start bytes_in=100 bytes_out=200
tactest -s 127.0.0.1 -k mykey -u myuser -acct stop bytes_in=400 bytes_out=300
tactest -s 127.0.0.1 -k mykey -u myuser -m 5 -acct stop bytes_in=400 bytes_out=300

Authorization Examples:
tactest -s 127.0.0.1 -k mykey -u myuser -author -service shell
tactest -s 127.0.0.1 -k mykey -u myuser -author -command configure terminal
tactest -s 127.0.0.1 -k mykey -u myuser -author -c 20 -command configure terminal

```

Figure 13: TACTest Help

5.2.1. TACTest Examples

The simplest method of running TACTest is to use it with just the shared secret (key), user, and user password.

```
C:\> tactest -k mykey -u myuser -p mypassword
```

For best results, explicitly define the server IP. This is important if you have multiple IP addresses on your computer. In some scenarios, the localhost IP (127.0.0.1) will not work, and you will need to manually define the server's routeable IP address in TACTest and/or in tacplus.xml.

```
C:\> tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword
```

To test the performance of your TACACS+ server, you can use the 'count' option.

```
C:\> tactest -s 127.0.0.1 -k mykey -u myuser -p mypassword -c 20
```

The Summary Statistics (below) will show you the transactions per second of your request. TACTest uses average possible transactions per second to extrapolate and estimate transactions per second based on the count option used above.

```

-----
SUMMARY STATISTICS
-----
Total Commands ..... 20
Successes ..... 20
Failures ..... 0
No Results ..... 0
Time Taken for commands ..... 0.02 secs
Avg Possible Transactions/Second ... 1000
Network Time per command ..... 0.0076 secs
Total Network time ..... 0.152 secs
Sent Transactions/Second ..... 444.4

```

Figure 14: TACTest Summary Statistics

For even better results, send 1,000 or more requests (-c 1000), send the request repeatedly for a specified amount of seconds (-m), send specified requests per second (-t), etc. You can send different types of requests like authorization requests (-type author) or commands (-type command). You can even use input files (-f filename) and output files (> filename) to store a log of your test.

6. Windows Firewall

A firewall is important for any production system. Because TACACS+ is a critical service-affecting service, you should enable the host firewall on the server even if you have a perimeter firewall.



You should disable any firewalls when doing your initial configuration and testing. Once you have confirmed your server is working as desired, and then enable the firewall before deploying in a production network.

The following example shows how to configure Windows Firewall on Windows XP. You may need to modify these settings slightly depending on your Operating System version.

1. Go to Start > Control Panel > Windows Firewall.
2. Under the General tab select "On" to enable the firewall.
3. Do not select "Don't allow exceptions".
4. Select the "Exceptions" tab.
5. Select "Add Port.."
6. For "Name:", enter TACACS+.
7. For "Port number:" enter 49.
8. Select "TCP".
9. Select "OK".
10. You should now see "TACACS+" with the checkbox selected under the Exceptions tab.
11. Select "OK".

7. Optimizing Performance

There are several steps you can take to optimize the performance of your server.

1. Disable any other services running on the local machine.
Other services may use resources differently than TACACS.net and can create additional load on the server, even if they are not servicing any requests. The services needed for general operation of Windows are fine, but you should not run any unnecessary or unneeded services. In addition to slowing down the server, this could also introduce unwanted security or troubleshooting complications.
2. Use IP addresses instead of hostnames in clients.xml.
When using hostnames, this requires an additional nslookup by the server, which could add additional latency to each request, and if your name server is unreachable, these lookups could fail. If you are going to use hostnames, you should be running DNS on the same server that is running TACACS.net.
3. Use Regular Expressions only when needed.
Regular expressions are a very powerful tool for configuring complex policies, but using them can potentially add slight latency to requests, so only use them when there is a good reason to do so. If you're not getting a lot of requests, there should be no difference in speed, but in a heavy use scenario, it could reduce performance somewhat.
4. Use Active Directory or local file groups for authentication.
The default group is a fallback group. For faster performance, use Active Directory or local file groups. When the server needs to lookup using local Users & Groups, it is not as fast and there could be a small performance hit. If the default group is left in the configuration, and the user is found in another group that is searched first in the file, the performance should not be affected, but this is not the ideal method.
5. Modify the caching period.
The <TimedCacheExpirySecs> option in tacplus.xml tells the server how long to cache Active Directory user credentials. The default setting is 60 seconds. If the username & password is cached, the server doesn't need to send a request to Active Directory so the server will respond to requests faster, but if the username or password changes during this period, the authentication may fail until the cache period expires. If usernames and passwords don't change often and you would like to speed up your server's response time, this setting may be increased.

60 seconds should be satisfactory for most purposes, so there should be no need to change this for most deployments.

8. Feature Enhancements

If you would like a feature or functionality in the TACACS.net TACACS+ Server that is not currently available, we can build it for you. Contact us through www.TACACS.net and give us the details on the feature(s) needed, the scenario how it would be used, your timeline and budget, and we will respond with a development quote to add the new feature.

9. Recommended Tools

Here are some of the tools that we recommend for use with the TACACS.net server.

1. MD5Sum
Command-line MD5 hash checker
<http://www.etree.org/md5com.html>
2. TerraTerm
Terminal emulator
<http://en.sourceforge.jp/projects/ttssh2/releases/>
3. WireShark
Protocol Analyser
<http://www.wireshark.org/>
4. Notepad++
<https://notepad-plus-plus.org/>
5. GNS3
Network simulator
<http://www.gns3.net/>