Interested in learning more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Implementing Secure Access to Cisco Devices using TACACS+ and SSH

Many environments that I encounter are using a Defense-In-Depth network security strategy. They have implemented firewalls, Intrusion Detection, VPN, and have a good security policy. When asked, however, how they manage their large installation of Cisco network devices, the reply many times is clear-text telnet, no username/password authentication combination, and very little in the way of auditing logs. The goal of this paper is to provide an easy guide for network administrators to implement secure remote access for ...

Copyright SANS Institute
Author Retains Full Rights

AD
Symantec™
Endpoint Protection 12
The next generation of reputation-based security
Download the beta
Symantec.
Confidence in a connected world.

Implementing Secure Access to Cisco Devices using TACACS+ and SSH

Paul Asadoorian, GCIA, GCIH
paul@pauldotcom.com
http://pauldotcom.com

Revision 1.2
May 13, 2003

© SANS Institute 2003, Author retains full rights

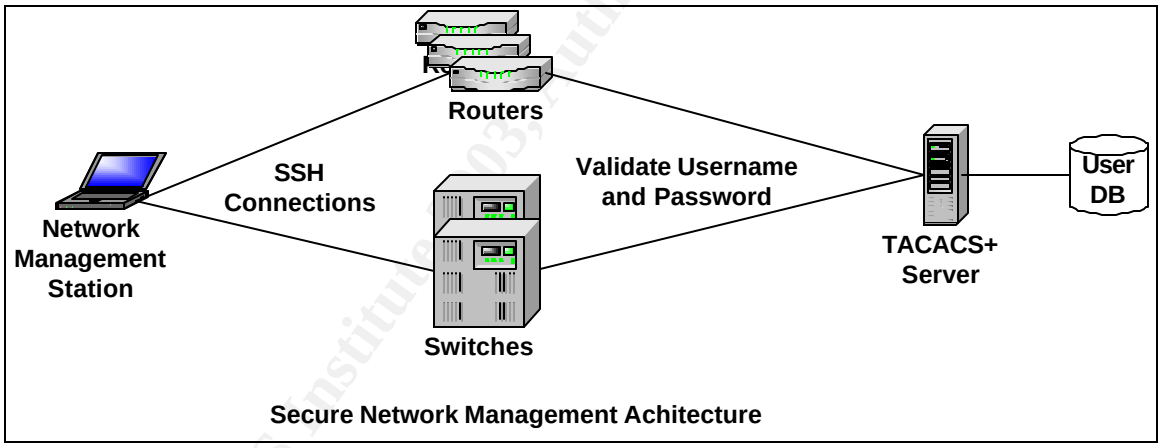
 Copyright © 2003, All Rights Reserved.

Goals

Many environments that I encounter are using a Defense-In-Depth network security strategy. They have implemented firewalls, Intrusion Detection, VPN, and have a good security policy. When asked, however, how they manage their large installation of Cisco network devices, the reply many times is clear-text telnet, no username/password authentication combination, and very little in the way of auditing logs. The goal of this paper is to provide an easy guide for network administrators to implement secure remote access for all Cisco networking equipment.

Topology and Supported Platforms

We will assume in this paper that the environment is using Cisco networking equipment, and running Cisco IOS version 12.1.1 or later with 3DES support. This covers switches (a limited selection) and routers right now, with support for access points and other devices on the way. The topology will be as follows:



The diagram above shows the network management station using SSH to connect to the networking equipment. This connection uses SSH protocol version 1 which provides for encryption. Be aware that SSH1, while encrypted, is vulnerable to Man-In-the-Middle attacks. See the "Further Recommendations" section for ways to help prevent against these attacks. SSH version 2 is not implemented on Cisco devices at the time of this writing.

Once a successful SSH connection is established you will need to enter a username and password. The network device takes the username and password and validates it with the TACACS+ server. The username is sent in clear-text, but the password is encrypted. The TACACS+ server receives the request and validates the username/password pair against a local UNIX user

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

database. The TACACS+ server then sends a message back to the network device indicating whether or not the username/password pair was in fact valid. This is a desirable setup because:

- The password is encrypted when traveling across the network throughout the entire process
- The primary authentication information (i.e. usernames and passwords) is not stored on the network device
- All devices point toward a central location, making it easy to change/expire users and passwords
- Accounting information is greatly enhanced since TACACS+ will log all commands that are entered into the device
- Authentication requires both a username and a password, which makes brute forcing the login more difficult than just a vty/enable password
- TACACS+ provides the ability to manage access at a very granular level; examples include using regular expressions to determine which commands a user can run.

TACACS+ Server configuration

The TACACS+ server discussed in this paper was written by Devrim Seral and can be downloaded from www.gazi.edu.tr/tacacs . It installs in similar fashion most other UNIX based software:

```
# tar zxvf tac_plus_v9a
# cd tac_plus_v9a
# ./configure
# make tac_plus
# make install
```

Once this has been compiled and installed, copy the default configuration to the /etc directory and open it up in your favorite editor. The following sections will go through each part of the configuration in detail:

```
##############################################################
# Default Config
##############################################################

# Key, very important
key = thisshouldbealongrandomstring

# Use /etc/passwd file to do authentication

default authentication = file /etc/passwd

# Accounting records log file
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
accounting file = /var/log/tacacs/tac_acc.log
```

The first configuration line is the "key" directive, which specifies the shared secret that will be used between all of your devices and the TACACS+ server. It needs to be the same on both the devices and the server in order for TACACS+ to work. The next line tells the TACACS+ server where to look for authentication, in this case we tell it to look to the local UNIX /etc/passwd file. You can also point your authentication at a RADIUS or LDAP server using additional modules. Finally we tell the TACACS+ server where to write the accounting logs, which we will setup to log command execution and logon/logoffs to all of our devices.

Next we will configure users and groups:

```
###########################################################
# Group Definitions
###########################################################

group = netadmin {
        default service = permit
        service = exec {
                priv-lvl = 15
        }
}

group = users {
        default service = deny
        service = exec {
                priv-lvl = 1
        }
}
```

To keep things simple we will use two groups, a privileged group and a non-privileged group. The "netadmin" group above will contain all of the network administrators who need enable access to the devices, and is granted privilege level 15, which is the highest level of access on Cisco devices (equivalent to *"root"* on a Unix-based system, or *"administrator"* on a windows-based system). When these users login they will be presented with the enable prompt immediately (Router#) and have the ability to enter privileged commands (i.e. "show running-config"). The group "users" will get privilege level 1, which only allows them to enter basic commands which are as follows:

```
access-enable    Create a temporary Access-List entry
clear            Reset functions
connect          Open a terminal connection
disable          Turn off privileged commands
disconnect       Disconnect an existing network connection
enable           Turn on privileged commands
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
exit              Exit from the EXEC
help              Description of the interactive help system
lock              Lock the terminal
login             Log in as a particular user
logout            Exit from the EXEC
name-connection   Name an existing network connection
ping              Send echo messages
rcommand          Run command on remote switch
resume            Resume an active network connection
show              Show running system information
ssh               Open a secure shell client connection
systat            Display information about terminal lines
telnet            Open a telnet connection
terminal          Set terminal line parameters
traceroute        Trace route to destination
tunnel            Open a tunnel connection
where             List active connections
```

WARNING: Although these are the only commands listed when querying the router in privilege level 1, the user has the ability to run other commands. For example in privilege level 1 you can run the "show ip route" and "show ip access-lists" commands. These alone give away critical information about the router, and more importantly the network topology. You can create custom filters to prevent users from running these commands, which we will cover in subsequent sections.

Next we will configure users:

```
##############################################################
# Netadmin users
##############################################################

user = bsmith {
        member = netadmin
}

##############################################################
# Unprivileged Users
##############################################################

user = sjones {
        member = users
        cmd = show {
                deny ip
                deny tacacs
                permit .*
        }
        cmd = quit {
                permit .*
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
        }
        cmd = exit {
                permit .*
        }
        cmd = logout {
                permit .*
        }
        cmd = ssh {
                permit 192\.168\.1\.[0-9]+
                deny .*
        }
}
```

In the first section above we place the user "bjones" in the netadmins group, which grants privilege level 15 on all the network devices. "bjones" also exists in the local Unix /etc/passwd file, which needs to be the case for all users whom you wish to grant access to via TACACS+. The next section is our unprivileged user "sjones", in which we grant privilege level 1. The users group in the TACACS+ server has a default deny statement, so by default no commands are allowed. We allow users to run almost every "show" command for debugging purposes, but do not let them see any IP or TACACS+ information. We also only let them SSH to machines on the 192.168.1.0/24 subnet, using a regular expression in the SSH section. Finally we do allow them to exit from the router, using any of the three commands that allow you to logoff. You will have to modify this section to fit your needs.

Once you have the configuration file adjusted to fit your needs you can start the TACACS+ server as follows:

```
# /usr/local/sbin/tac_plus -C /etc/tac_plus.cfg -d 248
```

The "-C" options tells the daemon where the configuration file is located, and the "-d 248" is the debugging level, which we set to 248, giving us plenty of information in the logs (see the tac_plus man page for more details).

You should create user accounts on this system and add them to the tac_plus.cfg file. Also, it is recommended that you have at least two TACACS+ servers and use rsync to synchronize the user accounts and TACACS+ configuration.

Configuring IOS

The configuration that we will step through in this section can be added to all of your IOS based network devices (primarily routers and switches). Be certain that you add the commands in the order they appear here, otherwise you can very easily lock yourself out of the device (not that I ever did that).

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

The first step is to setup the TACACS+ servers:

```
tacacs-server host 192.168.1.5
tacacs-server host 192.168.1.6
tacacs-server key thisshouldbealongrandomstring
```

The device will use the first server in the list if it is available, then the second, and so on. The key must be set to the same value as on the TACACS+ server.
The next command creates a local user, called "admin", with the privilege level of 15, and of course a good password:

```
username admin privilege 15 password agoodpasswordstring
```

This is the username/password pair that you will need to use if the TACACS+ server is unavailable. We need this local account in order to provide remote access via SSH only. Normally we default back to the enable password, but SSH requires a username and password pair, so the enable password does not work in this situation. Providing this local account will allow us to turn off telnet access to the device while still allowing access if the TACACS+ server is unavailable.

The AAA configuration is as follows:

```
aaa new-model
aaa authentication login default group tacacs+ local enable
aaa authorization exec default group tacacs+ local none
aaa authorization commands 0 default group tacacs+ local none
aaa authorization commands 1 default group tacacs+ local none
aaa authorization commands 15 default group tacacs+ local none
aaa accounting exec default start-stop group tacacs+
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+
```

The above configuration tells the device how to handle all interactive user logins and what the user can do once logged in. The first line simply creates a new AAA schema, allowing you to enter all of the commands that follow. The authentication line tells the device that whenever a user logs in it will validate the username and password against the TACACS+ server first, then look to a local username/password database, and finally default to the enable password. The only time it will default to the enable password is if the local username is not setup. The command and exec authorization works the same way, except if TACACS+ is not available the device will not perform any command or exec authorization. The last few lines will log all command activity on the device to the TACACS+ server. Next we will configure SSH.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

SSH should be configured on all your Cisco devices that will support it. The device will need to be running IOS version 12.1(1)T or later with 3DES to support this feature. To enable it, do the following:

```
ip domain-name mydomain.com
crypto key generate rsa
```

When you run the second command you will be prompted to enter a key size. Be certain to enter a key size of 1024 bits or greater. Once this is complete you will need to configure your terminal lines to use SSH:

```
line vty 0 15
     transport input ssh
```

The above configuration will disable telnet and only allow SSH connections to the router for remote access.

Further Recommendations

Below is a list of additional configuration that should be in place that complements the concepts described in this paper. Refer to "Securing Cisco Routers: A Step-By-Step Guide" for more detailed information on securing your Cisco IOS-based devices.

- All network devices should have an ACL that only allows network management workstations access to the device
- The TACACS+ server should be behind a firewall that only allows TACACS+ traffic (TCP port 49) in from all network devices.
- You can synchronize the user accounts easily using the following script:

```
#!/bin/sh
        while true; do
            for f in /etc/passwd /etc/shadow; do
                for h in second_auth_server_name ; do
                    rsync -azSHe ssh $f $h:$f
                done
            done
            sleep 60
        done
```

The above script needs to be run on your primary authentication server in the background. Be certain to replace the string "second_auth_server_name" with the hostname of your secondary authentication server and ensure that the proper hostname resolution is in place.

- You can synchronize the TACACS+ configuration using a similar script.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

- Use the port security feature and hard coded MAC addresses on switches and routers to help prevent against man-In-the-middle attacks that exploit ARP spoofing tactics.

Configurations

TACACS+ Server Configuration:

```
#################################################################
# Default Config
#################################################################

# Key, very important
key = thisshouldbealongrandomstring

# Use /etc/passwd file to do authentication

default authentication = file /etc/passwd

# Accounting records log file

accounting file = /var/log/tac_acc.log

#################################################################
# Group Definitions
#################################################################

group = netadmin {
        default service = permit
        service = exec {
                priv-lvl = 15
        }
}

group = users {
        default service = deny
        service = exec {
                priv-lvl = 1
        }
}

#################################################################
# Netadmin users
#################################################################

user = bsmith {
        member = netadmin
}
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
#################################################################
# Unprivileged Users
#################################################################

user = sjones {
        member = users
        cmd = show {
                deny ip
                deny tacacs
                permit .*
        }
        cmd = quit {
                permit .*
        }
        cmd = exit {
                permit .*
        }
        cmd = logout {
                permit .*
        }
        cmd = ssh {
                permit 192\.168\.1\.[0-9]+
                deny .*
        }
}
```

Cisco IOS Configuration:

```
tacacs-server host 192.168.1.5
tacacs-server host 192.168.1.6
tacacs-server key thisshouldbealongrandomstring
!
username admin privilege 15 password agoodpasswordstring
!
aaa new-model
aaa authentication login default group tacacs+ local enable
aaa authorization exec default group tacacs+ local none
aaa authorization commands 0 default group tacacs+ local none
aaa authorization commands 1 default group tacacs+ local none
aaa authorization commands 15 default group tacacs+ local none
aaa accounting exec default start-stop group tacacs+
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+
!
ip domain-name mydomain.com
crypto key generate rsa
!
line vty 0 15
transport input ssh
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
!
end
```

References

http://www.gazi.edu.tr/tacacs/ - TACACS+ Homepage, modified version of Cisco's freeware TACACS+ implementation

http://www.gazi.edu.tr/tacacs/docs/users_guide.txt - TAC_PLUS Developer's Kit vF4.0.2.alpha

http://www.cisco.com/warp/customer/707/ssh.shtml - Configuring Secure Shell on Routers and Switches Running Cisco IOS

http://www.cisco.com/en/US/tech/tk583/tk547/technologies_tech_note09186a008009465c.shtml - How to Assign Privilege Levels with TACACS+ and RADIUS

Securing Cisco Routers: Step-By-Step, Joshua L. Wright and John N. Stewart, SANS Press, August 2002 (http://store.sans.org/store_item.php?item=70&sans_store=6c8168578dfa75d9b21a67a028403792)

http://www.cisco.com/en/US/tech/tk583/tk617/technologies_security_advisory09186a00800b168e.shtml - Cisco Security Advisory: Multiple SSH Vulnerabilities

http://www.cisecurity.org/bench_cisco.html - The Center for Information Security Router Audit Tool (RAT), George Jones

Special Thanks

Joshua Wright, for editing this paper and encouraging me to write it, David Ball, Cisco TAC engineer who was especially helpful with the IOS configuration, and Don Wright for his help with testing the configurations and editing.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Last Updated: May 26th, 2013

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Malaysia @ MCMC 2013	**Cyberjaya, MY**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
SANS Pen Test Berlin 2013	**Berlin, DE**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
Industial Control Systems Security Training - Houston	**Houston, TXUS**	**Jun 10, 2013 - Jun 15, 2013**	**Live Event**
Security Impact of IPv6 Summit 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 16, 2013**	**Live Event**
SANSFIRE 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 22, 2013**	**Live Event**
SANS Canberra 2013	**Canberra, AU**	**Jul 01, 2013 - Jul 13, 2013**	**Live Event**
Digital Forensics & Incident Response Summit 2013	**Austin, TXUS**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS London Summer 2013	**London, GB**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS Rocky Mountain 2013	**Denver, COUS**	**Jul 14, 2013 - Jul 20, 2013**	**Live Event**
SANS Mumbai 2013	**Mumbai, IN**	**Jul 22, 2013 - Jul 27, 2013**	**Live Event**
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	**Washington, DCUS**	**Jul 22, 2013 - Jul 26, 2013**	**Live Event**
SANS San Francisco 2013	**San Francisco, CAUS**	**Jul 29, 2013 - Aug 03, 2013**	**Live Event**
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	**Bangalore, IN**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
SANS Boston 2013	**Boston, MAUS**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
Critical Security Controls Summit	**Washington, DCUS**	**Aug 12, 2013 - Aug 18, 2013**	**Live Event**
Industrial Control Systems Security Training - DC	**Washington, DCUS**	**Aug 12, 2013 - Aug 16, 2013**	**Live Event**
SANS Thailand 2013	**Bangkok, TH**	**Aug 19, 2013 - Aug 31, 2013**	**Live Event**
SANS Virginia Beach 2013	**Virginia Beach, VAUS**	**Aug 19, 2013 - Aug 30, 2013**	**Live Event**
Mobile Device Security Summit 2013	**OnlineCAUS**	**May 30, 2013 - Jun 06, 2013**	**Live Event**
SANS OnDemand	**Books & MP3s OnlyUS**	**Anytime**	**Self Paced**