

Troubleshooting Cisco Access Registrar

Revised: March 20, 2009, OL-17222-03

This chapter provides information about techniques used when troubleshooting Cisco Access Registrar (CAR) and highlights common problems.

Gathering Basic Information

Table 26-1 lists UNIX commands that provide basic and essential information to help you understand the CAR installation environment.

Table 26-1 UNIX Commands to Gather Information

UNIX Command	Information Returned
<code>/usr/bin/uname -r</code>	
<code>/usr/bin/uname -i</code>	
<code>/usr/bin/uname -v</code>	
<code>/usr/bin/uname -a</code>	All system information including hostname, operating system type and release, machine model and type
<code>/usr/sbin/prtconf</code>	
<code>/usr/sbin/df -k</code>	File system disk space usage including partitions, capacity, and space used
<code>/usr/bin/ps -ef</code>	Currently running processes
<code>/usr/sbin/psinfo -v</code>	Information about processors
<code>/usr/bin/pkginfo -l CSCCOar</code>	



Note

More information about these commands and their options is available using the **man** terminal window on the Sun workstation.

Troubleshooting Quick Checks

Many of the most common problems can be diagnosed by doing the following:

-
-
-

Disk Space

- Failure to process RADIUS requests
- Parts of the CAR configuration *disappearing*

```
$INSTALL /tmp
```

Resource Conflicts

Resource conflicts are a common reason for the Cisco Access Registrar 4.1 server failing to start. The most common resource conflicts are the following:

- Cisco Network Registrar is running on the CAR server
- Another application is also using ports 1645 and 1646
 - A network management application is using the Sun SNMP Agent

No Co-Existence With Cisco Network Registrar

Cisco Network Registrar cannot coexist on a machine running CAR for this reason. You can determine if CNR is running by entering the following command line in a terminal window:

```
pkginfo | grep -i "network registrar"
```

Port Conflicts

```
netstat -aP tcp
```

```
netstat -aP udp
```



Server Running Sun SNMP Agent

If you plan to use the CAR server's SNMP agent, you cannot use the Sun Microsystems SNMP agent that comes with the Solaris operating system.

Cisco Access Registrar Log Files

Examining the CAR log files can help you diagnose most CAR issues. By default, the CAR log files are located in `/var/log/cisconf`. [Table 26-2](#) lists the CAR log files and the information stored in each log.

CAR Log Files

Log File	Information Recorded
<code>agent_server_1_log</code>	
<code>ar-status</code>	<code>arserver</code>
<code>aregcmd_log</code>	
<code>config_mcd_1_log</code>	
<code>name_radius_1_log</code>	
<code>name_radius_1_trace</code>	<code>aregcmd</code>

Modifying File Sizes for Agent Server and MCD Server Logs

```
car.conf          $BASEDIR/conf          agent_server_logs
config_mcd_server_logs logs
```

```
AGENT_SERVER_LOG_SIZE (10 MB by default)
```

```
AGENT_SERVER_LOG_FILES (2 by default)
```

You will find these new parameters at the beginning of the `agent_server_logs` file. When the log file size reaches the value set in `AGENT_SERVER_LOG_SIZE`, a rollover of the file occurs. The value set in `AGENT_SERVER_LOG_FILES` specifies the number of log files to be created.

Using xtail to Monitor Log File Activity

A useful way of monitoring all of the log files is to run `xtail` on the `agent_server_logs` file.

```
xtail
```

`xtail $INSTALL/logs/*`



Modifying the Trace Level

- Level 0—No tracing occurs
- Level 1—Indicates when a packet is sent or received and when a status change occurs in a remote server (RADIUS Proxy and LDAP)
- Level 2—Information includes the following:
 - Which client and vendor objects are being used to process a packet
 - More details about remote servers (RADIUS Proxy and LDAP), packet transmission, and timeouts
 - Details about poorly-formed packets.

Level 3—Information includes the following:

- Tracing of errors in Tcl scripts when referencing invalid RADIUS attributes
- Which scripts have been run
- Details about local userlist processing

Level 4—Information includes the following:

- Advanced duplication detection processing
- Details about creating, updating, and deleting sessions
- Tracing of all APIs called during the running of a script

Level 5—Provides information about policy engine operations

Installation and Server Process Start-up

-
-
-

>

```
AR RADIUS server running      (pid: 6285)
AR MCD lock manager running   (pid: 6284)
AR MCD server running        (pid: 6283)
AR Server Agent running      (pid: 6277)
```

aregcmd and Cisco Access Registrar Configuration

[<server>] [<level>]

When you do not specify a server, CAR sets the trace level for all servers in the current cluster. When you do not specify a trace level, the currently set level is used. The default trace level is 0.

CAR can be in two states, running or stopped. In either state, all four CAR processes remain running. The state of CAR will be displayed when logging into or by using the command:

```
Server 'Radius' is Running, its health is 10 out of 10\
```

```
Stopping Server 'Radius'...
Server 'Radius' is Stopped
```

```
Starting Server 'Radius'...
Server 'Radius' is Running, its health is 10 out of 10
```

```
Reloading Server 'Radius'...
Server 'Radius' is Running, its health is 10 out of 10
```

-
-
-
-
-

RADIUS Request Processing

-

down

Other Troubleshooting Techniques and Resources

aregcmd Stats Command

```
Global Statistics for Radius:
serverStartTime = Tue Oct  2 10:28:02 2008
serverResetTime = Tue Oct  2 20:25:12 2008
serverState = Running
totalPacketsInPool = 1024
totalPacketsReceived = 0
totalPacketsSent = 0
totalRequests = 0
totalResponses = 0
totalAccessRequests = 0
totalAccessAccepts = 0
totalAccessChallenges = 0
totalAccessRejects = 0
totalAccessResponses = 0
totalAccountingRequests = 0
totalAccountingResponses = 0
totalStatusServerRequests = 0
totalAscendIPAAlocateRequests = 0
totalAscendIPAAlocateResponses = 0
totalAscendIPAReleaseRequests = 0
totalAscendIPAReleaseResponses = 0
totalUSRNASRebootRequests = 0
totalUSRNASRebootResponses = 0
totalUSRResourceFreeRequests = 0
totalUSRResourceFreeResponses = 0
totalUSRQueryResourceRequests = 0
totalUSRQueryResourceResponses = 0
totalUSRQueryReclaimRequests = 0
totalUSRQueryReclaimResponses = 0
totalPacketsInUse = 0
totalPacketsDrained = 0
totalPacketsDropped = 0
totalPayloadDecryptionFailures = 0
```

```
core:          ELF 32-bit MSB core file SPARC Version 1, from 'radius'
```

assertion

Chapter 11, “Using Replication.”