CCNP Routing and Switching TSHOOT 300-135 Quick Reference

Brent Stewart

Cisco Press

CCNP

Routing and Switching TSHOOT 300-135 Quick Reference

Brent Stewart

Cisco Press
800 East 96th Street
Indianapolis, Indiana 46240 USA

CCNP Routing and Switching TSHOOT 300-135 Quick Reference

Brent Stewart

Published by:
Cisco Press
800 East 96th Street
Indianapolis, IN 46240 USA

Printed in the United States of America

First Printing November 2014

ISBN-13: 978-0-13-392948-5

ISBN-10: 0-13-392948-5

Warning and Disclaimer

This book is designed to provide information about networking. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Publisher
Paul Boger

Associate Publisher
Dave Dusthimer

Business Operation Manager, Cisco Press
Jan Cornelssen

Executive Editor
Brett Bartow

Managing Editor
Sandra Schroeder

Development Editor
Marianne Bartow

Senior Project Editor
Tonya Simpson

Copy Editor
Paula Lowell

Technical Editor
Sean Wilkins

Editorial Assistant
Vanessa Evans

Cover Designer
Mark Shirar

Composition
Studio Galou

Indexer
Brad Herriman

Proofreader
Megan Wade-Taxter

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact international@pearsoned.com.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
168 Robinson Road
#28-01 Capital Tower
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Europe Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: +31 0 800 020 0791
Fax: +31 0 20 357 1100

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

About the Author

Brent Stewart, CCNP, CCDP, CCSI, MCSE, is the vice president of Managed Services at Stalwart Systems (stalwartsystems.com), an innovative IT engineering firm focused on secure IT architectures. His experience includes designing and managing a large-scale worldwide voice, video, and data network. He was a course director for Global Knowledge and participated in the development of BSCI with Cisco and has written and taught extensively on CCNA and CCNP. Brent lives in Hickory, North Carolina, with his beautiful wife, Karen, and their mischievous children Benjamin, Kaitlyn, Madelyn, and William.

About the Technical Reviewer

Sean Wilkins is an accomplished networking consultant for SR-W Consulting and has been in the field of IT since the mid-1990s, working with companies such as Cisco, Lucent, Verizon, and AT&T, as well as several other private companies. Sean currently holds certifications with Cisco (CCNP/CCDP), Microsoft (MCSE), and CompTIA (A+ and Network+). He also has a Master of Science in information technology with a focus in network architecture and design, a Master of Science in organizational management, a Master's Certificate in network security, a Bachelor of Science in computer networking, and Associates of Applied Science in computer information systems. In addition to working as a consultant, Sean spends most of his time as a technical writer and editor for various companies; check out his work at his author website: www.infodispersion.com.

Contents at a Glance

Contents

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([ ]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

How This Book Is Organized

- **Chapter 1, "Tools and Methodologies of Troubleshooting":** This chapter focuses on minimizing time-to-repair. It examines the techniques that can be applied to decrease downtime. The scientific method is suggested as a model for troubleshooting. Descriptions of tasks commonly used to maintain performance and prepare for problems, such as documentation and scheduled preventative maintenance, are provided. Finally, it covers IOS tools, such as archiving, logging, and configuration rollback, that are valuable in the troubleshooting process.
- **Chapter 2, "Troubleshooting Switching Technologies":** Ethernet is ubiquitous in campus networks and data centers. More and more services are traveling on Ethernet, such as storage, virtualization, and telephony. This chapter describes troubleshooting the critical pieces: Spanning Tree, VLANs, InterVLAN routing, and gateway redundancy.
- **Chapter 3, "Troubleshooting IP Networking":** This chapter describes issues around IP services and starts with a discussion of IP addressing. It also discusses services such as NTP, syslog, and SNMP.
- **Chapter 4, "Troubleshooting Routing Technologies":** This chapter covers troubleshooting link layer connectivity, OSPF, EIGRP, and BGP routing protocols and router performance.

This page intentionally left blank

CHAPTER 3

Troubleshooting IP Networking

IP Address Assignment

This section describes issues regarding IP services and starts with a discussion about IP addressing. Even when Layer 2 connectivity is intact, failures at Layer 3 prevent communication.

IP addresses identify interfaces, not devices. The principal IP parameters are address, subnet mask, and (except on routers) default gateway. Addresses are unique, but a mask and gateway should be common on a subnet. Address information can be either statically assigned or obtained from Dynamic Host Configuration Protocol (DHCP).

When troubleshooting IP connectivity, refer to the following steps:

Step 1. Use **traceroute** to test connectivity or (if broken) the last point working in the path.

Step 2. If the problem is in the local subnet, ensure that the IP address is unique. Non-unique addresses create a log message that looks like "1w2d: %IP-4-DUPADDR: Duplicate address 10.1.1.1 on FastEthernet0/1, sourced by 0002.1234.5678." Also ensure that the subnet mask is the same as the one on the gateway, and that the gateway IP is correct. Many organizations use the first usable address in the subnet for the default gateway to prevent such confusion.

Step 3. If the problem is between the local subnet and the endpoint, verify whether routes exist on the intermediate devices. It is easy to forget to troubleshoot the return path, but make sure you check this as well. If the far end doesn't know how to route back, the symptoms will be identical to an outbound routing problem.

Step 4. Check intermediate devices for logic that would interfere with routing, such as policy-based routing, access lists, and network address translation. If possible, remove these elements for testing.

Addresses can also be assigned through DHCP. DHCP is sometimes used for network devices, particularly on commodity connections to service providers. More commonly, DHCP is used on end systems, and troubleshooting from a network perspective involves understanding how DHCP is supported by the router.

DHCP passes configuration information, including IP configuration, upon request. A client sends a broadcast DHCPDiscover to ask for information and the server responds with DHCPOffer. Because more than one DHCP server could respond, the client responds to the offer it is accepting with DHCPRequest. Most clients accept the first offer received. Routers can either act as DHCP servers or as relays, which forward the broadcasts to remote servers.

A simple DHCP configuration is shown in Example 3-1. It supplies DHCP addresses in the range 10.1.1.0/24 and runs on the interface attached to that subnet. Verify the current assigned addresses using **show ip dhcp binding** or **show ip dhcp pool**. You can also see the action of DHCP by using **debug ip dhcp server packet|event**.

Example 3-1 DHCP

```
Ip dchp pool tshoot
  Network 10.1.1.0 255.255.255.0
  Default-router 10.1.1.1
  Dns-server 10.1.1.3
```

Supporting a remote DHCP server requires forwarding broadcast DHCPDiscover messages to a remote server. For instance, a branch core switch might forward DHCP requests back to a central resource using the helper address, as shown in the following configuration:

```
Interface vlan100
  Ip helper-address 10.1.1.1
```

DHCP events can create a disproportionate amount of drama because of the impact on users. The principal way to recognize a DHCP problem is that PCs have addresses that start 169.254. These are auto-assigned addresses used when no DHCP server is found. DHCP problems break into three large groups.

Start-up problems are issues where network connectivity starts slowly and is not in place until after the DHCP process times out. A perfect example is when spanning-tree startup places a port in blocking. Start-up issues can be recognized because rerunning DHCP works after the port is active (on Windows, this can be tested using **winipcfg/renew**). Spanning-tree startup can be fixed by applying PortFast.

Connectivity failures to the server also prevent DHCP from completing as expected. In this second type of problem, PCs also auto-assign an address but will not be fixable by renewing later. Troubleshoot this by following the path back to the DHCP server and verifying whether DHCP traffic can flow. DHCP servers can fail, and a best practice is to run multiple DHCP servers within an enterprise. A failed server is also verified (and worked around) by temporarily placing a DHCP server on the local router.

The third group of DHCP problems is the ignorant or malicious introduction of an unauthorized DHCP server in the network. Surprise! If a consumer device is brought in, such as a wireless access point, it can be recognized by 192.168.1.0/24 addresses being assigned. When a rogue DHCP server is introduced maliciously, detecting it can be difficult. An attacker could do this to assign legitimate addresses with a traffic capture device listed as the gateway. When recognized, the ARP table can be used to track back the switch port of the bogus gateway.

DHCP snooping is a technique to deal with spurious DHCP offers. An administrator identifies a port that has an authorized DHCP server (this works best if the DHCP server is the only device attached through that port). DHCP snooping then drops DHCPOffers coming from untrusted ports. It also drops release messages coming from ports other than the port where a user was assigned a given address. Violations also get logged, such as %DHCP_SNOOPING-5-DHCP_SNOOPING_UNTRUSTED_PORT.

Configure DHCP snooping globally and identify a trusted interface as shown in the following configuration. View DHCP snooping information by using **show ip dhcp snooping**.

```
DSW1(config)# ip dhcp snooping
DSW1(config)# interface f1/1
DSW1(config-if)# ip dhcp snooping trust
```

NTP, Syslog, and SNMP

NTP, Syslog, and SNMP are services that aid in troubleshooting. Each needs to be set up beforehand so that data can be used in the troubleshooting process. Network Time Protocol (NTP) helps establish causality—which event came first—when comparing logs from different devices. Syslog allows log collection in a central location. Simple Network Management Protocol (SNMP) collects network telemetry in a central location. SNMP can gather data points, such as capacity utilization, processor and memory usage, and error rate. From the SNMP collector, these data points can be graphed to show sudden changes or to analyze trends and suggest proactive next steps.

NTP

Example 3-2 is a standard NTP setup. NTP starts by identifying a time zone. NTP draws time from a time server on the Internet—clock1.unc.edu in this example—and acts as a time server (master) for the network. Two important pieces to this configuration are to protect time records from changes meant to obfuscate an attack (time communication is encrypted to prevent malicious changes to time records) and to communicate time to known devices only.

Example 3-2 NTP Configuration

```
clock timezone EST -5
clock summer-time CDT recurring
clock calendar-valid
ntp update-calendar

ntp master 2        !stratum 2
ntp authenticate
ntp authentication-key 11 md5 tshoot
ntp trusted-key 11

access-list 1 permit clock1.unc.edu
access-list 1 deny any
access-list 2 permit R2
access-list 2 deny any

ntp access-group peer 2
ntp access-group serve-only 1

ntp server clock1.unc.edu
ntp peer R2
```

Verify NTP using **show ntp associations**. In Example 3-3, a public NTP server is specified and synched. If a server does not sync, it is almost always because of a communication issue. Troubleshoot by verifying whether the server is reachable with **ping** and that NTP traffic is not blocked by a firewall or access list.

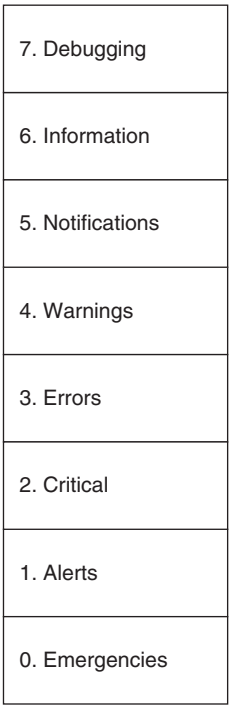
Example 3-3 NTP Associations

```
R1# show ntp associations
address             ref clock         st      when    poll   reach
 delay   offset    disp
*~152.2.21.1        152.2.21.1        1       29      1024   377
 4.2     -8.59     1.6
* master (synced), # master (unsynced), + selected, - candidate,
  ~ configured
```

Syslog

Logging errors to the system is enabled by default, but it must be enabled to ship the logs to a remote destination. The logging level must be specified as well. Figure 3-1 illustrates levels 0 through 7. Logging at least at level 3 and making sure that the time on the device is accurate are good practices.

Figure 3-1 Logging Levels



Logging issues are generally related to routing; however, the local device keeps a copy of its logs in memory and can be referenced if logs are not flowing to the central server. A generic configuration for logging is shown in the following configuration:

```
Service timestamps log datetime
Logging trap 3
```


SNMP

Most of TSHOOT concerns active break/fix situations, but the most common problems are usually the slowly developing, insidious type. A good example is when capacity utilization creeps up. By the time the default network monitoring system (users) recognize there is an issue, it is too late! Ordering and

receiving new service takes months, during which time service will be unsatisfactory. Enterprises of any size should have some system to gather SNMP telemetry and analyze it regularly. A number of commercial and open-source packages are quite good.

SNMP is also a way for an attacker to control a network, so it must be used cautiously, kept updated, and secured. A sample SNMP configuration is shown in Example 3-4.

Example 3-4 SNMP Access List

```
Snmp-server community tsh0ot RO
Snmp-server community S3cret RW 10
Snmp-enable traps
Snmp-server host 10.1.1.3
Ip access-list 10 permit host 10.1.1.3
```

Two communities are defined—one is read-only and one allows administrative changes. The read/write string is limited to the defined host.

Gateway Redundancy

Hosts are configured with a default gateway—a router address that passes traffic off the local subnet. The problem is that router failures strand the hosts. The solution is first-hop redundancy protocols, which enable two routers to cooperatively support a single IP, which is then given to hosts as a default gateway.

The three first-hop redundancy protocols are as follow:

- Hot Standby Router Protocol (HSRP) is an older Cisco proprietary protocol. One router is the active and one is the standby. The routers pass keepalives that enable the standby to recognize failure of the primary router.
- Virtual Router Redundancy Protocol (VRRP) is an open standard but is otherwise similar to HSRP. Because HSRP works, many organizations have continued to use it.
- Gateway Load Balancing Protocol (GLBP) is an open standard, but it enables simultaneous load balancing over as many as four gateways.

Because HSRP is the most common, this section focuses on HSRP. The general configuration and troubleshooting strategy applies as well to VRRP and GLBP, however.

HSRP is configured under the interface using standby commands. Routers in the same HSRP group share a MAC address and IP address, so **standby** is used to identify the group and virtual IP.

By default, each HSRP speaker has a priority of 100. The speaker with the highest priority is the active router. If a new router starts, however, HSRP does not change the active router until the failure of the active router. To change this so that the higher priority is instantly recognized, use the **preempt** command. Example 3-5 shows an HSRP snippet to illustrate the configuration.

Example 3-5 Example HSRP

```
Interface f0/0
  Ip address 10.1.1.2 255.255.255.0
  Standby 2 ip 10.1.1.1
  Standby 2 priority 120
  Standby 2 preempt
```

Verify the HSRP state of a router using **show standby**, which summarizes this information to a table as shown in Example 3-6. To see detailed information on HSRP, such as timers and virtual MAC, use **show standby** (as shown in this example).

Example 3-6 HSRP

```
R1# show standby
GigabitEthernet0/1 - Group 135
  State is Active
    23 state changes, last state change 25w6d
  Virtual IP address is 135.159.64.1
  Active virtual MAC address is 0000.0c07.ac87
    Local virtual MAC address is 0000.0c07.ac87 (v1 default)
  Hello time 5 sec, hold time 20 sec
    Next hello sent in 0.284 secs
  Preemption enabled
  Active router is local
  Standby router is unknown
  Priority 150 (configured 150)
  Group name is "hsrp-Gi0/1-135" (default)
Richardson-rtr01# show standby interface gi0/1
Global           Confg: 0000
Gi0/1 If hw      BCM1125 Internal MAC (27), State 0x210040
Gi0/1 If hw      Confg: 0000
Gi0/1 If hw      Flags: 0000
Gi0/1 If sw      Confg: 0000
```

```
Gi0/1 If sw      Flags: 0000
Gi0/1 Grp 135    Confg: 0072, IP_PRI, PRIORITY, PREEMPT, TIMERS
Gi0/1 Grp 135    Flags: 0000

HSRP virtual IP Hash Table (global)
103 172.25.96.1    Gi0/1      Grp 135

HSRP MAC Address Table
43 Gi0/1 0000.0c07.ac87
    Gi0/1 Grp 135
```

Just as **show standby brief** provides an HSRP overview, **show vrrp brief** and **show glbp brief** describe the VRRP and GLBP environments, respectively. Similarly, **show standby interface** and **debug standby** have equivalents for the other first-hop redundancy protocols.

A simple configuration for HSRP might look like the following:

```
Interface f0/0
  Ip address 10.1.1.2 255.255.255.0
  Standby 43 ip 10.1.1.1
```

Troubleshoot first-hop redundancy protocols by using the following steps:

Step 1. Make sure that the "real" address of the routers is unique. Two routers in a standby group each have a different permanent address. In the previous simple configuration, the interface is addressed as 10.1.1.2/24.

Step 2. Check that the standby IP and group numbers match. In the previous configuration, the interface is in HSRP group 43 and the shared IP is 10.1.1.1.

Step 3. Verify that the standby IP address is in the local subnet. If the shared address isn't in the local subnet, local devices won't be able to reach it! In the earlier configuration, 10.1.1.1 is in the 10.1.1.0/24 subnet.

Step 4. Verify whether access lists are not filtering standby traffic. Access lists must permit keepalive traffic between the participating routers and allow clients to use both routers as a gateway.

Filtering

Access lists (ACL) are a tool to block traffic as it crosses a router. Standard IP access lists filter traffic based on source address. Extended access lists

filter traffic based on source and destination IP address, as well as transport layer details.

Note

Filter traffic based on destination address using a route to null0. To filter traffic going to 192.168.21.0/24, enter a static route: **ip route 192.168.21.0 255.255.255.0 null0**.

Standard IP access lists are numbered 1–99 or named. Extended access lists are numbered 100–199 or named. Using named access lists is strongly encouraged because this aids in understanding the intent of the ACL later. Access lists are applied to interfaces in a direction (in or out, relative to the device), and access lists always end with an implicit deny.

A simple standard access list might be used on the perimeter to block bogus source addresses (known as *bogons*) as illustrated in Example 3-7. Assuming that fastethernet1/1 is the outside interface, the access list needs to be applied inbound (traffic coming into the router from outside would have source addresses that should be blocked).

Example 3-7 Bogon List

```
Interface fastethernet1/1
  Ip access-group bogon_list in
Ip access-list standard bogon_list
  Deny 0.0.0.0 0.255.255.255
  Deny 10.0.0.0 0.255.255.255
  Deny 127.0.0.0 0.255.255.255
  Deny 169.254.0.0 0.0.255.255
  Deny 172.16.0.0 0.15.255.255
  Deny 192.168.0.0 0.0.255.255
  Deny 224.0.0.0 31.255.255.255
  Permit any
```

A slightly more complicated configuration might be used to limit access from a "guest" network as shown in Example 3-8. The idea here is that guests are allowed to attach through fastethernet1/2 (perhaps this is attached to a wireless access point). Guests should be allowed DNS and web access, but nothing else. Note that the access list allows tcp/80 and udp/53—everything else is denied implicitly.

Example 3-8 Guest ACL

```
Interface fastethernet1/2
  Ip access-group guest_access in
Ip access-list extended guest_access
```

```
Permit tcp any any eq 80
Permit udp any any eq 53
```

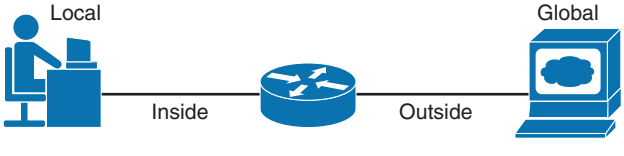
Focus on four principal areas for access list troubleshooting:

1. Is this an ACL problem? If possible, remove the access list and test.
2. Check the applied interface and direction. The temptation is to review the access list first, but verifying direction is important so that sources and destinations are correctly interpreted. It is also possible that the applied interface is not on the path the traffic is taking!
3. Check the access list entries. Are addresses and port numbers entered correctly?
4. Review the order of operations. Remember that ACLs are processed top-down and the first match (permit or deny) is executed. Could the traffic be matching an earlier statement? Check this by moving the line in question to the top of the list temporarily.

NAT

Network Address Translation (NAT) is the process of translating source or destination IP addresses as traffic traverses a router. Router interfaces are described as outside (the public side) or inside (the private side). Translations are described from points of view—*local* describes the address seen by an inside observer and *global* describes what a public observer sees, as shown in Figure 3-2.

Figure 3-2 NAT Terms



NAT can map addresses to pool or to a single address. Mapping all the inside connections to a single public address is an example of Port Address Translation (PAT) or overloading.

Steps to Troubleshooting NAT

Consider the following steps to troubleshooting NAT. Each step is explored further in the sections that follow.

Step 1. Attempt to rule out NAT. Check connectivity.

Step 2. Understand the objective of NAT.

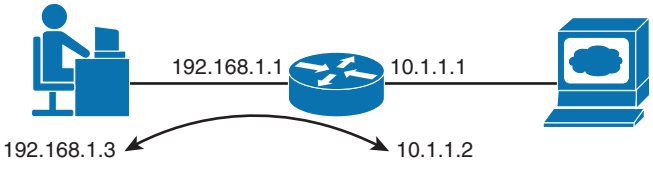
Step 3. Verify the translation.

Step 4. Verify whether the translation is being used.

Rule Out NAT

From the translating router, source a **ping** from the outside interface as shown in Figure 3-3. This verifies that traffic flows bidirectionally from the router and that neighbors have a return route. If **ping** doesn't work, check the connectivity from the router. If all lines appear stable, then investigate routing, especially the return path.

Figure 3-3 Simple NAT



As shown in Figure 3-3, use an extended **ping** or **traceroute** to originate traffic from the 10.1.1.1 interface to check connectivity. If possible, examine neighboring devices to determine whether their routing tables include the return route to 10.1.1.0/24.

Understand the Objective of NAT

Review what the NAT configuration is trying to accomplish, and verify that all elements of the configuration appear correct. A generic configuration for NAT identifies inside and outside interfaces and an **ip nat** command as shown in Example 3-9. A common error is to either not identify or to misidentify interfaces.

Example 3-9 Simple NAT 1:1 Configuration

```
Interface f0/0
  Ip add 10.1.1.1 255.255.255.0
  Ip nat outside
Interface f0/1
  Ip address 192.168.1.2 255.255.255.0
  Ip nat inside
Ip nat inside source static 192.168.1.3 10.1.1.2
```

In Example 3-9, which builds on Figure 3-3, the router is being asked to translate traffic from the user to a single outside address. Note that the inside and outside are correctly identified and that the **ip nat** statement has the addresses in the correct order.

Verify the Translation

Make sure that the translation is entered into the translation table on the router correctly. This is accomplished by using **show ip nat translation**. We are translating the inside address, so the following configuration tells us that the inside address, as seen from the Internet, is 10.1.1.2. The inside address, as seen from the private network, is 192.168.1.3.

```
R2# show ip nat translation
Pro Inside global     Inside local      Outside local     Outside global
--- 10.1.1.2          192.168.1.3        ---               ---
```

Verify the Translation Is Being Used

Having checked the configuration and the translation table, the last step is to ensure that the traffic is actually being translated. This is accomplished by using **debug ip nat**. Example 3-10 shows the debug output. Here, the PC is pinging an outside address to create traffic.

Example 3-10 debug NAT

```
R2# debug ip nat
R2# show log
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
       Console logging: level debugging, 39 messages logged
       Monitor logging: level debugging, 0 messages logged
       Buffer logging: level debugging, 39 messages logged
       Trap logging: level informational, 33 message lines logged

Log Buffer (4096 bytes):

13:53:33: NAT: s=192.168.1.3->10.1.1.2, d=10.1.1.100[70]
13:53:33: NAT*: s=10.1.1.100, d=10.1.1.2->192.168.1.3 [70]
```

Some less common errors can occur in translation, but examining the system log as shown in the preceding calls out these issues as well.

Authentication

Authentication verifies whether legitimate users are accessing the device. CCNA covers simple authentication, such as the enable password. In enterprise networks, using "box passwords" becomes difficult to manage. Passwords become old and can't easily be changed when administrators leave the company.

Centralized authentication is therefore important as networks grow. Centralized authentication is commonly accomplished through TACACS+ or RADIUS servers (IOS also supports Kerberos). Multiple authentication checks can be specified—these are checked in order, checking subsequent sources only if the preceding method times out.

When you're configuring authentication, specify a method name (in Example 3-11 this is *widgets*). If the method name is the default, then the method is applied to all appropriate interfaces on the router. Example 3-11 creates a widgets method and applies it to telnet sessions. Widgets checks an ACS server using TACACS, and if that times out, widgets refers to the local username and password.

Example 3-11 Authentication

```
Aaa new-model
Aaa authentication login widgets group tacacs+ local
Tacacs-server host 10.1.1.3
Tacacs-server key 5tandardW1dg3t5
Username admin password tshoot
Line vty 0 4
  Login authentication widgets
```

Troubleshooting Authentication, Authorization, and Accounting (AAA) involves three important checks:

1. Verify whether the server is working. Can other devices authenticate using the server?
2. Confirm whether the server is reachable. Can you **ping** the server? Is the shared secret correct?
3. Verify whether the credentials are spelled correctly and are still valid.