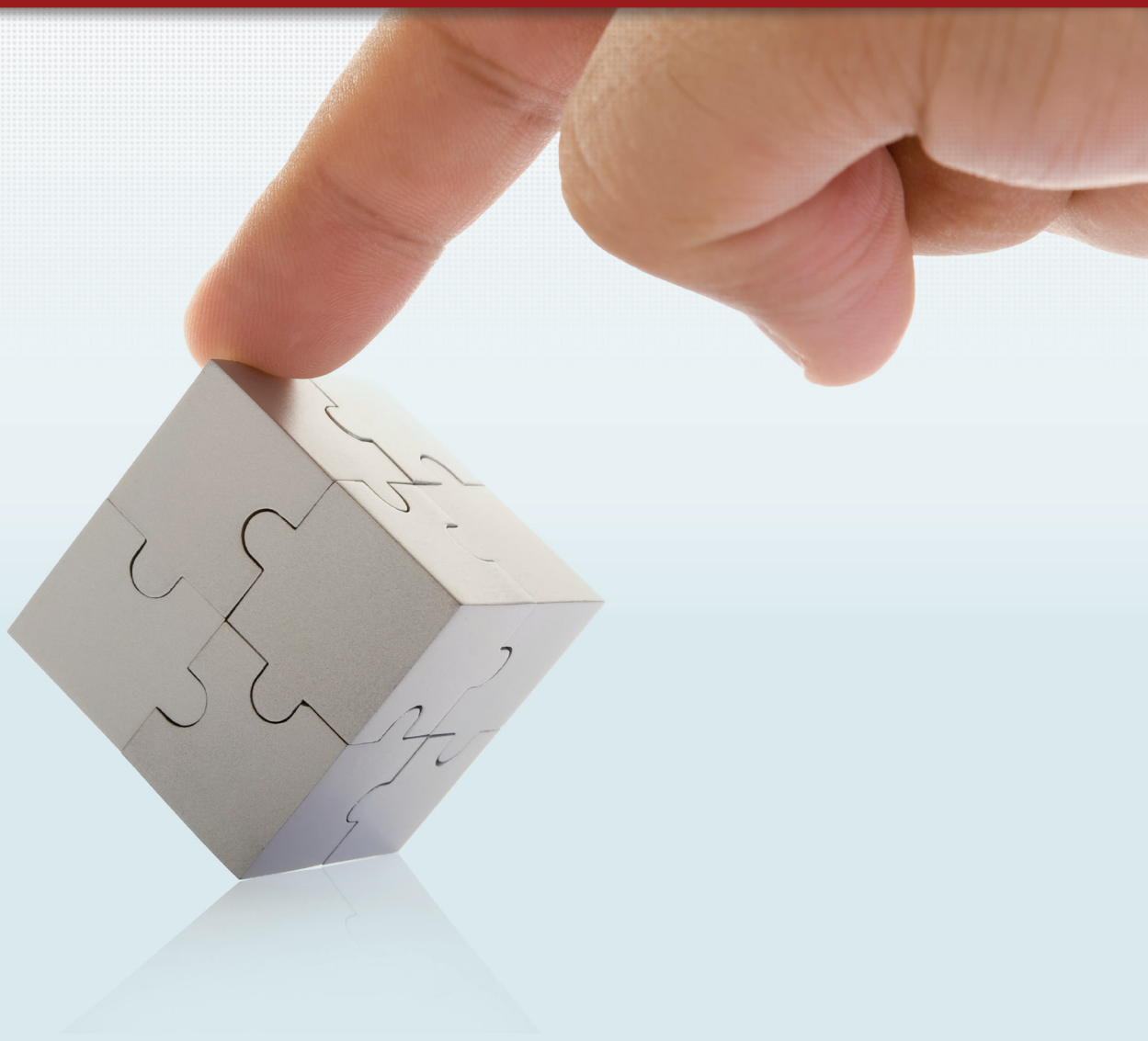
ipexpert

IPexpert's

Network Security Operation and Troubleshooting



Authored By: Anthony Sequeira CCIE# 15626 (R&S), CCDP, CCSP.
Technical Editor: Carl Yost Jr CCIE# 30486 (R&S), Jason Gooley CCNP
Editor: Tiffany Pagan

Before We Begin

This product is part of the IPexpert suite of materials that provide CCIE candidates and network engineers with a comprehensive training program. For information about the full solution, contact an IPexpert Training Advisor today.

Telephone: +1.810.326.1444
Email: sales@ipexpert.com

Congratulations! You now possess one of the ULTIMATE CCIE™ Lab preparation and network operation resources available today! Senior engineers, technical instructors, and authors boasting decades of internetworking experience produced this resource.

In order to enjoy technical support from IPexpert and your CCIE community, be sure to visit the following Internet resources:

http://blog.ipexpert.com
http://onlinestudylist.com

IPexpert is proud to lead the industry with multiple support options at your disposal free of charge. Our online communities have attracted a membership of over 20,000 of your peers from around the world! At blog.ipexpert.com, you can keep up to date with everything IPexpert does and read the latest in technical articles from world-renowned IPexpert instructors. At OnlineStudyList.com, you may subscribe to multiple "SPAM-free," moderated CCIE-focused email lists.

Feedback

Do you have a suggestion or other feedback regarding this book or other IPexpert products? At IPexpert, we look to you – our valued clients – for the real world, frontline evaluation that we believe is necessary so that we may always improve. Please send an email with your thoughts to feedback@ipexpert.com or call 1.866.225.8064 (international callers dial +1.810.326.1444).

In addition, for those using this book as CCIE™ preparation, when you pass the CCIE™ Lab exam, we want to hear about it! Email your CCIE™ number to success@ipexpert.com and let us know how IPexpert helped you succeed. We would like to send you a gift of thanks and congratulations.

Additional CCIE™ Preparation Material

IPexpert, Inc. is committed to developing the most effective Cisco CCIE™ R&S, Security, Voice and Wireless Lab certification preparation tools available. Our team of certified networking professionals develops the most up-to-date and comprehensive materials for networking certification, including self-paced workbooks, online Cisco hardware rental, classroom training, online (distance learning) instructor-led training, audio products, and video training materials. Unlike other certification-training providers, we employ the most experienced and accomplished teams of experts to create, maintain, and constantly update our products. At IPexpert, we are focus on making your CCIE™ Lab preparation more effective.

Issues with this Book

This book is carefully edited to ensure the accuracy of all content. Should you find any error whatsoever, please email a page reference and detailed comment to compsolv@me.com. Your email will be responded to promptly.

IPEXPERT END-USER LICENSE AGREEMENT

END USER LICENSE FOR ONE (1) PERSON ONLY

IF YOU DO NOT AGREE WITH THESE TERMS AND CONDITIONS,
DO NOT OPEN OR USE THE TRAINING MATERIALS.

This is a legally binding agreement between you and IPEXPERT, the “Licensor,” from whom you have licensed the IPEXPERT training materials (the “Training Materials”). By using the Training Materials, you agree to be bound by the terms of this License, except to the extent these terms have been modified by a written agreement (the “Governing Agreement”) signed by you (or the party that has licensed the Training Materials for your use) and an executive officer of Licensor. If you do not agree to the License terms, the Licensor is unwilling to license the Training Materials to you. In this event, you may not use the Training Materials, and you should promptly contact the Licensor for return instructions.

The Training Materials shall be used by only ONE (1) INDIVIDUAL who shall be the sole individual authorized to use the Training Materials throughout the term of this License.

Copyright and Proprietary Rights

The Training Materials are the property of IPEXPERT, Inc. ("IPEXPERT") and are protected by United States and International copyright laws. All copyright, trademark, and other proprietary rights in the Training Materials and in the Training Materials, text, graphics, design elements, audio, and all other materials originated by IPEXPERT at its site, in its workbooks, scenarios and courses (the "IPEXPERT Information") are reserved to IPEXPERT.

The Training Materials cannot be used by or transferred to any other person. You may not rent, lease, loan, barter, sell or time-share the Training Materials or accompanying documentation. You may not reverse engineer, decompile, or disassemble the Training Materials. You may not modify, or create derivative works based upon the Training Materials in whole or in part. You may not reproduce, store, upload, post, transmit, download or distribute in any form or by any means, electronic, mechanical, recording or otherwise any part of the Training Materials and IPEXPERT Information other than printing out or downloading portions of the text and images for your own personal, non-commercial use without the prior written permission of IPEXPERT.

You shall observe copyright and other restrictions imposed by IPEXPERT. You may not use the Training Materials or IPEXPERT Information in any manner that infringes the rights of any person or entity.

Exclusions of Warranties

THE TRAINING MATERIALS AND DOCUMENTATION ARE PROVIDED "AS IS." LICENSOR HEREBY DISCLAIMS ALL OTHER WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW THE LIMITATION OF INCIDENTAL DAMAGES OR LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU. This agreement gives you specific legal rights, and you may have other rights that vary from state to state.

Choice of Law and Jurisdiction

This Agreement shall be governed by and construed in accordance with the laws of the State of Michigan, without reference to any conflict of law principles. You agree that any litigation or other proceeding between you and Licensor in connection with the Training Materials shall be brought in the Michigan state or courts located in Port Huron, Michigan, and you consent to the jurisdiction of such courts to decide the matter. The parties agree that the United Nations Convention on Contracts for the International Sale of Goods shall not apply to this License. If any provision of this Agreement is held invalid, the remainder of this License shall continue in full force and effect.

Limitation of Claims and Liability

ANY ACTION ON ANY CLAIM AGAINST IPEXPERT MUST BE BROUGHT BY THE USER WITHIN ONE (1) YEAR FOLLOWING THE DATE THE CLAIM FIRST ACCRUED, OR SHALL BE DEEMED WAIVED. IN NO EVENT WILL THE LICENSOR'S LIABILITY UNDER, ARISING OUT OF, OR RELATING TO THIS AGREEMENT EXCEED THE AMOUNT PAID TO LICENSOR FOR THE TRAINING MATERIALS. LICENSOR SHALL NOT BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, REGARDLESS OF WHETHER LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. WITHOUT LIMITING THE FOREGOING, LICENSOR WILL NOT BE LIABLE FOR LOST PROFITS, LOSS OF DATA, OR COSTS OF COVER.

Entire Agreement

This is the entire agreement between the parties and may not be modified except in writing signed by both parties.

U.S. Government - Restricted Rights

The Training Materials and accompanying documentation are “commercial computer Training Materials” and “commercial computer Training Materials documentation,” respectively, pursuant to DFAR Section 227.7202 and FAR Section 12.212, as applicable. Any use, modification, reproduction release, performance, display, or disclosure of the Training Materials and accompanying documentation by the U.S. Government shall be governed solely by the terms of this Agreement and shall be prohibited except to the extent expressly permitted by the terms of this Agreement.

IF YOU DO NOT AGREE WITH THE ABOVE TERMS AND CONDITIONS, DO NOT OPEN OR USE THE TRAINING MATERIALS AND CONTACT LICENSOR FOR INSTRUCTIONS ON RETURN OF THE TRAINING MATERIAL

Contents

Chapter 1: Introduction to Network Security Operation and Troubleshooting

Chapter 1: Introduction to Network Security Operation and Troubleshooting introduces the team of authors, consultants, and editors that completed this book and describes the book's purpose. This chapter also provides suggestions for the usage of this written work. This introductory chapter also covers a basic overview of network security operation and troubleshooting concerns.

About the Authors

Anthony Sequeira, CCIE No. 15626 (R&S), formally began his career in the information technology industry in 1994 with IBM in Tampa, Florida. He quickly formed his own computer consultancy, Computer Solutions, and then discovered his true passion—teaching and writing about Microsoft and Cisco technologies. Anthony is currently pursuing his second CCIE in the area of Security, and is a full-time instructor for the next generation of KnowledgeNet: www.StormWindLive.com. He recently achieved his VMware Certified Professional certification. When Anthony is not writing or lecturing about the latest innovations in networking technologies, you may find him flying a Cessna in the Florida skies.

About the Technical Editors

Carl Yost Jr., CCIE No. 30486 (R&S), currently works as a Network Engineer/Director of I.T. for a health care company in Buffalo NY. He has worked in numerous roles in I.T. since 1998. Carl is currently preparing for the CCIE in Security while living with his wife and children in Western New York. When not surrounded by Cisco devices, Carl truly enjoys working with Redhat Linux.

Jason Gooley, CCNP, Jason is a highly motivated network engineer with over 17 years of experience in the communications industry. Based in Chicago, Jason currently manages the network for the nation's most famous next day carpet company. Jason is currently in the process of pursuing his CCIE certification for Routing and Switching while also expanding his knowledge in Unified Communications and Security.

About the Editor

Tiffany Pagan began her career in editing in 1997. Throughout her career, she has worked with several private individuals and companies such as Moffitt Cancer Center and Tampa General Hospital. Tiffany is currently working on writing her own series of short stories as well as working as an editor and personal assistant. Tiffany resides in Tampa, Florida with her husband and three beautiful children.

Who Should Read this Book?

This text has two primary audiences. The first audience is for those CCIE candidates that are searching for the most comprehensive and error-free materials available for the operation and troubleshooting of key technologies presented in the various tracks of the CCIE written and practical lab exams. These students should possess a home rack of equipment for CCIE-level command-line practice, they should possess an equipment emulator, or they should rent equipment from a company like www.proctorlabs.com. The authors and technical editors exhaustively tested all of the demonstrations found throughout the text and the important end of chapter Trouble Ticket challenges against all practice rack options described earlier. Where issues arise with popular equipment emulators, the text makes note. This book is the most remarkably thorough and technically accurate book written on the subject of network security to date.

The book's second audience is those readers that must support network security technologies in their actual production network environments. This book serves as an amazing guide and reference for real-world problem solving within production networks that deploy these specific technologies. In fact, while many courses and texts purport to have certification success as a by-product of a thorough investigation of all protocols, this book actually succeeds in this approach.

How to Use this Book

This book breaks specific network security technologies down on a chapter-by-chapter basis for a complete and thorough review of this broad set of topics. Each chapter begins with a review of the selected technology. Following this, the text provides an intense examination of the operation of the protocols, including key aspects of troubleshooting for the specific technology. After this, the chapter presents some of the most common issues that can result with a particular technology, and most importantly, details the simple troubleshooting tools and steps that succeed for remediation.

Each chapter concludes with sample troubleshooting scenarios that provide a full walkthrough of a well-designed approach for troubleshooting each major issue. The text provides reference guides for the most popular and powerful **show** and **debug** commands for a specific technology.

Some chapters also contain sample Trouble Tickets on specific technologies. Readers may download initial configurations for these sample Trouble Tickets, or install them in a simple Graphical User Interface (GUI) on www.proctorlabs.com. These sample Trouble Tickets allow students to build confidence and expertise by actually troubleshooting issues in the network security domain presented in the chapter.

Students are encouraged to follow along on a rack of equipment for every section of every chapter. This really enhances and strengthens the learning process.

An Introduction to Network Security

Open any major periodical these days and you will find many articles about network security and the damage done to corporations or government entities. Cisco realizes this of course and ensures that their latest network devices are equipped with the most modern security tools and technologies. The idea behind this is to create a defense in depth approach. To add layers of security for the overall network by taking advantage of the security capabilities that exist in each device.

For example, a client device connects to a wireless access point. The wireless access point uses the latest in wireless security mechanisms to protect the overall network. The wireless access point connects to a Layer 2 switch. This switch is also equipped with powerful security mechanisms. This Layer 2 switch connects to a multilayer switch. Once again, this device is configured with the latest in security technologies. That device connects to a Layer 3 router…you know of course there is excellent security technologies in place there.

While this defense in depth strategy sounds excellent, notice that for it to really work, there must be a careful configuration of these various security mechanisms across the various devices. That is where this book comes in! Enjoy this no-nonsense approach to configuring some of the most important network security features available today.

While many will enjoy this text chapter by chapter, here is a handy reference for the material found in each chapter should you need to jump right to a particular technology:

Contents At A Glance

VI. Chapter 6: Cisco IOS and Zone-Based Firewalls

   a. Basic Zone-Based Firewall

   b. Deep Packet Inspection

VII. Chapter 7: NAT

   a. Static NAT

   b. Dynamic NAT

   c. Static PAT

   d. Policy-Based NAT

VIII. Chapter 8: Other Security Features

   a. TCP Intercept

   b. Fragment Attacks

   c. Antispoofing with ACLs

   d. uRPF

   e. Telnet

   f. SSH

   g. IOS IPS

Chapter 2: AAA and Security Server Protocols

It is impossible to consider yourself proficient in network security without knowing the authentication, authorization, and accounting (AAA) system and related protocols very well. In this chapter, you will master this material and be able to describe its use and operation on Cisco devices, as well as troubleshoot issues related to AAA.

AAA and Security Server Protocols Technology Review

Local Privilege Authentication and Authorization

When you first begin your education about Cisco devices and how they operate, you quickly learn about the local database approach to authorization. Let us review the key points here.

There are four main methods for administrative access to the Cisco router or switch that you can control using all local techniques. These administrative access options are as follows:

- The console port (**con 0**)
- The virtual terminal lines or vty ports (**vty 0 4**)
- The auxiliary port (**aux 0**)
- HTTP (**ip http server**)

The weakest method of protecting the device is with local password-only checking. This is accomplished as follows:

```
R1(config)#line con 0
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#end
R1#
```

Notice this is very straightforward, but beginning students are often confused by the **login** command. This command does not permit logins and they suspect. This command is instructing the device to check the local password configured under the line upon login. As a result, it surprises the students that if they type **no login** under the line, it not only permits logins, but it does so with no password checking.

Obviously, it is a much better idea to protect authentication against the device using a username and password combination. While an employee in your network might gladly give a peer a password not attached to them in any way, this employee might think twice before giving a peer their own username and password!

Follow this simple procedure to create username and password entries in the local database:

```
R1(config)#username JOHNS secret cisco
R1(config)#username AMYS secret cisco
R1(config)#end
R1#
```

Once the local database is populated, using it for authentication is simple. Notice the following examples:

```
R1(config-line)#exit
R1(config)#line con 0
R1(config-line)#login local
R1(config-line)#exit
R1(config)#ip http authentication local
R1(config)#end
R1#
```

From an authorization standpoint, remember that there are privilege levels on the Cisco device. Specifically, there are 16 levels of access. These levels are numbered 0 to 15. Level 0 is not really accessible – commands at this level are automatically accessible to all users. Level 1 is the privilege level you experience when you are in user mode. Level 15 is what you experience when you are in privileged mode (many call this enable or enabled mode).

For many years, the approach to custom levels of authorization locally on the device was to literally take advantage of the different levels of privilege on the device. For example:

```
R1(config)#privilege exec level 2 configure terminal
R1(config)#privilege configure level 2 hostname
R1(config)#end
R1#
```

Here the commands of **configure terminal** and **hostname** are moved to level 2. This level can now be assigned to a user account as follows:

```
R1(config)#username BRIANS privilege 2 secret cisco
R1(config)#end
R1#
```

Notice that when this user logs in – they only have access to those commands, and the commands that are available at lower privilege levels:

```
User Access Verification

Username: BRIANS
Password:
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#?
Configure commands:
  beep      Configure BEEP (Blocks Extensible Exchange Protocol)
  call      Configure Call parameters
  default   Set a command to its defaults
  dss       Configure dss parameters
  end       Exit from configure mode
  exit      Exit from configure mode
  help      Description of the interactive help system
  hostname  Set system's network name
  netconf   Configure NETCONF
  no        Negate a command or set its defaults
```

```
  oer       Optimized Exit Routing configuration submodes
  sasl      Configure SASL

R1(config)#
```

A much more popular, flexible, and friendly method of doing local authorization now is to use custom views of the command line environment. The first step is to enable CLI views:

```
R1(config)#enable secret cisco
R1(config)#aaa new-model
R1(config)#exit
R1#
enable view
Password:
R1#
*Mar  1 00:50:10.815: %PARSER-6-VIEW_SWITCH: successfully set to view 'root'.
R1#
```

Once you have enabled views and you are in this 'root' view – it is time to create a new custom view of the CLI. For example:

```
R1(config)#parser view VIEW_MYVIEW
R1(config-view)#
*Mar  1 00:53:12.995: %PARSER-6-VIEW_CREATED: view 'VIEW_MYVIEW' successfully created.
R1(config-view)#secret cisco
R1(config-view)#commands exec include show access-list
R1(config-view)#commands exec include configure terminal
R1(config-view)#commands configure include ip access-list extended
R1(config-view)#commands ipenacl include all deny
R1(config-view)#commands ipenacl include all permit
R1(config-view)#end
R1#
```

This view can now be associated with a username and will apply when they log in:

```
R1(config)#username BELLAS view VIEW_MYVIEW secret cisco
R1(config)#end
R1#
```

AAA Server Authentication and Accounting with TACACS+

Remember, AAA relies upon a security protocol when it comes time for communication between the AAA client (a router or other such device) and the AAA server (for example, Cisco Access Control Server (ACS). One powerful option is the Terminal Access Controller Access Control System Plus (TACACS+) protocol.

There are many facts about the TACACS+ protocol that you should commit to memory if you are interested in Cisco Routing and Switching or Security certifications – these facts are:

- TACACS+ is a Cisco proprietary security protocol that might not be supported by all networking vendors

- The protocol relies upon the Transmission Control Protocol; it utilizes TCP port 49 in its operation; remember, TCP means reliability mechanisms are used in the network communication
- A shared secret key is used between the client and the sever - a combination of a hashing algorithm and an XOR function; MD5 to hash the shared secret key
- Each portion of AAA is performed separately under TACACS+ – this is not the case as you will see with RADIUS
- TACACS+ is considered more secure than RADIUS; this is because there is encryption and authentication of the entire message

When dealing with TACACS+ authentication, three packets are used. They are START, REPLY, and CONTINUE. The REPLY can indicate - ACCEPT, REJECT, ERROR, CONTINUE.

Under TACACS+ authorization, there are two message types – REQUEST and RESPONSE. The RESPONSE can contain:

- FAIL
- PASS_ADD - authorized and additional information returned
- PASS_REPL - REQUEST ignored and additional information returned
- FOLLOW - authorization should take place on other server
- ERROR - most common is secret key mismatch

Attribute-Value (AV) pairs determine the authorized services.

With TACACS+ accounting the process is very similar to authorization. There is a use of AV pairs. There is also a START record, a STOP record, and a CONTINUE record. REQUEST and RESPONSE messages are also used. The RESPONSE can indicate SUCCESS, ERROR, or FOLLOW.

AAA Server Authentication and Accounting with RADIUS

Another option for a security protocol for the communication between the AAA client and the AAA server is the Remote Authentication Dial-In User Service (RADIUS). Here are the facts that you should commit to memory!

- RADIUS is an open standard defined in RFC 2865
- RADIUS uses UDP; 1645/1812 for authentication and 1646/1813 for accounting
- Authentication and authorization performed together, while accounting is done separately
- RADIUS is Extensible - vendors can add new attribute values
- The accounting functions of RADIUS are considered superior to TACACS+
- Only the password is encrypted in RADIUS, therefore TACACS+ is considered more secure

RADIUS uses a shared secret key to authenticate the RADIUS messages. A key message type is the Access-Request.

Using AAA

Examine this sample AAA configuration. Notice how a list of authentication methods is applied to a certain administrative access area. Notice also how there is a default list of authentication methods for cases where a custom list is not provided:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#aaa new-model
R1(config)#aaa authentication login default local
R1(config)#aaa authentication login LIST_TELNET tacacs+ local
R1(config)#line vty 0 4
R1(config-line)#login authentication LIST_TELNET
R1(config-line)#exit
R1(config)#username JOHNS secret cisco
R1(config)#tacacs-server host 10.1.1.100
R1(config)#tacacs-server key cisco
R1(config)#end
R1#
```

As you can imagine, the RADIUS configuration is nearly identical. One very important note about the operational behavior of this configuration is that if the user fails the TACACS+ authentication, the local fallback method is not used. What is this fallback method for then? It is a fallback method in the event that the TACACS+ server(s) is not available.

Common Issues with AAA and Security Server Protocols

When in a CCIE lab exam environment for Routing and Switching, issues with AAA and the security server protocols of TACACS+ and RADIUS are going to be limited to misconfigurations on the router or switch for the desired behavior. It is critical that you recall all options for local authentication as well as AAA remote options. Let's review those options here – then enjoy sample Trouble Tickets:

- Line password configuration
- Local username and password configuration
- Authorization using privilege levels
- Authorization using command line views
- AAA using remote servers and TACACS+
- AAA using remote servers and RADIUS

For most troubleshooting scenarios – simple **show run** commands should be more than adequate. Also, keep in mind these points:

- In remote AAA server environments, a frequent requirement is a "backdoor" configuration that allows access via the local database for those entryways where a specific list of methods is not assigned. This is accomplished with the **aaa authentication login default local** command. Notice that the keyword local can be substituted with the appropriate method.
- In extreme cases of complex issues (most typically real world deployments), the **debug aaa authentication** command can be helpful in pinpointing issues.
- With AAA authentication lists that you create custom, ensure they are properly assigned to the appropriate entry method.
- Watch for filters that might block either the RADIUS or TACACS+ protocols.

Chapter Challenge: AAA and Security Server Protocols Trouble Tickets

The following section includes sample trouble tickets for AAA and Security Server Protocols. Remember to load the appropriate initial configurations for these sample tickets. Here is the simple topology that will be used for these tickets:

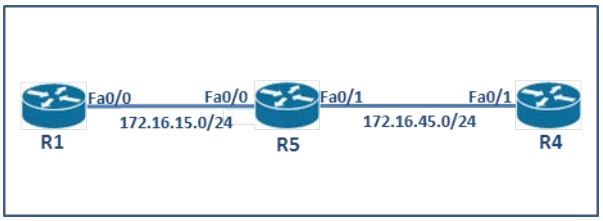


Figure 2-1 A Sample Troubleshooting Topology

Trouble Ticket #1

Your supervisor has indicated that he wants R1 to provide authentication of a new junior administrator when this administrator attempts to access the device using Telnet. The username should be **TIMOTHY** and the password should be **12@ttReDDwen**. This password should not be visible in clear text on the device. Do not use the **service password-encryption** command in this configuration.

2 points

Trouble Ticket #2

Your supervisor has indicated that he wants a default AAA login policy in place on R5 that ensures the enable password of **cisco** is in use on the device. For Telnet access, he wants a custom list named TELNETACCESSCUSTOM. This should use a TACACS+ server at 10.10.10.100 with a key of **cisco123!**. Fallback should be to use the local username and password database.

3 points

Chapter Challenge: AAA and Security Server Protocols Trouble Ticket Solutions

The following section includes solutions to the Trouble Tickets associated with AAA and security server protocols.

Trouble Ticket #1

Your supervisor has indicated that he wants R1 to provide authentication of a new junior administrator when this administrator attempts to access the device using Telnet. The username should be **TIMOTHY** and the password should be **12@ttReDDwen**. This password should not be visible in clear text on the device. Do not use the **service password-encryption** command in this configuration.
2 points

Step 1 - Fault Verification:

What is the current authentication policy for Telnet access on R1?

```
R1#show run | begin vty 0 4
line vty 0 4
 password 12@ttReDDwen
 login
!
!
end
```

The current configuration is checking for the correct password for Telnet login, but it is not challenging the user for a username. This should be easy to fix with the correct commands. I notice also that the password is correct in the current configuration so I can copy and paste this for accuracy.

Step 2 – Fault Remediation

Here is the correct configuration for this scenario:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#username TIMOTHY secret 12@ttReDDwen
R1(config)#line vty 0 4
R1(config-line)#login local
R1(config-line)#end
R1#
```

There are two things to note with this configuration:

- If you decide to copy and paste the password from the existing configuration for accuracy, be very careful that you do not copy an extra space at the end of the password. This extra space will count as a character and cause failure for the task as the password will be incorrect.

- Be sure to use the command username secret in order to populate the local database. This ensures the password will not appear in clear text without the use of the **service password-encryption** command.

Step 3 – Fault Remediation Verification

Here, we can move to R5 and ensure that the Telnet access to R1 is authenticated in the correct manner:

```
R5#telnet 172.16.15.1
Trying 172.16.15.1 ... Open


User Access Verification

Username: TIMOTHY
Password:
R1>exit

[Connection to 172.16.15.1 closed by foreign host]
R5#
```

Trouble Ticket #2

Your supervisor has indicated that he wants a default AAA login policy in place on R5 that ensures the enable password of **cisco** is in use on the device. For Telnet access, he wants a custom list named TELNETACCESSCUSTOM. This should use a TACACS+ server at 10.10.10.100 with a key of **cisco123!**. Fallback should be to use the local username and password database.

3 points

Step 1 - Fault Verification:

What is the current default AAA login policy?

```
R5#show run | include default
R5#
```

There is no default policy in place on the device. In remediation, I will also ensure the correct enable secret is in place.

Is there the correct custom list in place?

```
R5#show run | include authentication
aaa authentication login TELNETACCESSCUSTOM group tacacs+ none
R5#
```

Notice there is a list in place, but the list is not correct.

We also discover that there is no TACACS+ server configured on the device:

```
R5#show run | include tacacs-server
R5#
```

Finally, we check for any filter that might be in place and discover one:

```
R5#show run | include access-group
 ip access-group AL_FILTER out
R5#show access-list
Extended IP access list AL_FILTER
    5 deny ip 172.16.45.0 0.0.0.255 host 10.10.10.100
    10 permit ip 172.16.45.0 0.0.0.255 any
R5#
```

Step 2 – Fault Remediation

Here is the correct configuration for the default AAA method portion of the scenario. Notice that solving this in sections can help improve our accuracy:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#aaa authentication login default enable
R5(config)#enable secret cisco
```

```
R5(config)#
```

Here is the next solution portion:

```
R5(config)#aaa authentication login TELNETACCESSCUSTOM group tacacs local
R5(config)#line vty 0 4
R5(config-line)#login authentication TELNETACCESSCUSTOM
R5(config-line)#end
R5#
```

Notice that we must remember to assign this custom list to the correct area.

Now it is time to configure the TACACS+ server connectivity:

```
R5(config)#tacacs-server host 10.10.10.100 key cisco123!
R5(config)#
```

Finally, let us fix the filter to ensure that TACACS+ server connectivity is not broken:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#ip access-list extended AL_FILTER
R5(config-ext-nacl)#no 5
R5(config-ext-nacl)#end
R5#
```

Step 3 – Fault Remediation Verification

Here the verification involves a series of show commands as follows:

```
R5#show run | include default
aaa authentication login default enable
R5#show run | include TELNETACCESSCUSTOM
aaa authentication login TELNETACCESSCUSTOM group tacacs+ local
 login authentication TELNETACCESSCUSTOM
R5#show run | include tacacs-server
tacacs-server host 10.10.10.100 key cisco123!
R5#show access-list
Extended IP access list AL_FILTER
    10 permit ip 172.16.45.0 0.0.0.255 any
R5#
```

Chapter 3: Access Lists

There are many types of access lists that you must master on the Cisco devices. This chapter covers the most important and often used forms. These are areas that are guaranteed to be tested at many levels of Cisco Certification.

Access List Technology Review

As an author of yet another important text on networking topics, I am always compelled to point out the most important chapter. I certainly think that for this book, this might be it. As a network administrator you will often need Access Control Lists (ACLs) to stop certain forms of traffic from entering an important area of your network. When we use ACLs in this manner, we are calling upon their ability to ***filter*** traffic. In fact, using your Cisco router to filter certain traffic forms while permitting others to flow through turn your router into a ***firewall*** device. Admittedly, the router is a very basic firewall with this implementation. We refer to it as a stateless or static packet firewall. Later in this text, you will learn how to introduce a much more robust firewall solution for the router termed the Zone Based Firewall. This provides the ability to engage in sophisticated stateful firewalling and even some advanced application inspection.

ACLS might sound rather simplistic, and they actually are. While the syntax can get a bit complex to master at first, the operation of these mechanisms is actually quite simple. The ACL inspects network packets based on a wide variety of criteria. There are simple access lists (termed standard) that inspect packets based on source address only, and more sophisticated access lists (termed extended) that examine packets based on much more criteria including source and destination address, port numbers, etc.

In addition to the important role of filtering traffic in our network, access control lists can also be called upon to classify traffic forms. For example, you might use an access control list to define the exact "interesting traffic" that you want to prioritize in a particular Quality of Service (QoS) mechanism.

As you might guess, the focus of this chapter will indeed be upon the use of ACLs to filter traffic. Specifically, the use of ACLs to permit or deny:

- Packets entering or exiting specific interfaces
- Traffic attempting to pass through the router
- Telnet traffic entering or exiting the Virtual Terminal Lines (VTY) ports for router administration

Note: It is very important to realize in the interest of troubleshooting that thanks to ICMP messaging returned by Cisco routers when dropping packets due to ACLs, a Destination Unreachable (U.U.U) will be seen in response to a ping and an Administratively Prohibited (!A * !A) will be seen in response to a traceroute.

Inbound and Outbound ACL Operation

It is critical to remember (again in the interest of troubleshooting), that ACLs do not impact traffic generated by the router itself. For example, if you create an ACL that clearly denies Telnet traffic, and you apply this ACL outbound on a Serial interface of the router, the ACL has no impact when you try and Telnet from that router. What about an ACL tied to the VTY lines of the router using the **access-class** command? Again, the ACL will not affect a user logged in locally on the router trying to telnet out, but a user that has specifically Telnetted into the router will not be able to Telnet out.

Remember that ACLs operate in two possible ways:

- Inbound ACLs - incoming packets are processed before they are routed to an outbound interface. Notice how efficient these ACLs can be for the router. The router does not need to worry about the routing process for packets that are not to be forwarded anyways.
- Outbound ACLs - incoming packets are routed to the outbound interface, and then they are processed through the outbound ACL.

The ACL as a List of Tests

Remember that the word list in Access Control List means quite a bit. The ACL is indeed an ordered list of statements. It is critical that you understand how the router processes this ordered list of statements:

- The ordered statements in the ACL are checked, in order, for a match against the packet.
- If there is an ACL statement match, the rest of the statements in the list are skipped.
- The packet is then permitted or denied as determined by the matched statement.
- If the packet does not match an ACL statement, the packet is tested against the next statement in the list.
- This matching process continues until the end of the list is reached.
- A final implicit deny all statement ends the list and causes all packets not matching an explicit entry to be dropped.

Note: A common and embarrassing error for engineers when first working with ACLs is to construct an ACL that consists of all deny statements. You can imagine their surprise when ***all*** packets are dropped by an interface as a result.

Types of IPv4 ACLs

As described earlier, standard ACLs are able to match on source IP address. Notice how this results in the permitting or denying of all protocols in the IP suite based on the source of traffic. You can use a name or a number to identify the ACL, and these methods are now functionally equivalent in the router IOS. If you are using a number to identify the ACL, standard ACLs are numbered 1 to 99, or 1300 to 1999.

Let us review the syntax possible for the standard ACL. Notice that I am using a numbered approach in this example:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list ?
  <1-99>            IP standard access list
  <100-199>         IP extended access list
  <1100-1199>       Extended 48-bit MAC address access list
  <1300-1999>       IP standard access list (expanded range)
  <200-299>         Protocol type-code access list
  <2000-2699>       IP extended access list (expanded range)
  <700-799>         48-bit MAC address access list
  dynamic-extended  Extend the dynamic ACL absolute timer
  rate-limit        Simple rate-limit specific access list

R1(config)#access-list 10 ?
```

```
  deny    Specify packets to reject
  permit  Specify packets to forward
  remark  Access list entry comment

R1(config)#access-list 10 permit ?
  Hostname or A.B.C.D  Address to match
  any                  Any source host
  host                 A single host address

R1(config)#access-list 10 permit 192.168.1.0 ?
  A.B.C.D  Wildcard bits
  log      Log matches against this entry
  <cr>

R1(config)#access-list 10 permit 192.168.1.0 0.0.0.255 ?
  log  Log matches against this entry
  <cr>

R1(config)#access-list 10 permit 192.168.1.0 0.0.0.255 log ?
  <cr>

R1(config)#access-list 10 permit 192.168.1.0 0.0.0.255 log
R1(config)#
```

Note: Another common (and embarrassing) mistake with ACLs is creating them perfectly and then not applying them to an interface. Just ask CCIE-great Scott Morris about that one. Scott was the only candidate to succeed in the construction of a very complex ACL in his lab exam attempt, only to lose full points for not applying it.

Here is a review of the syntax for applying an ACL to an interface:

```
R1(config)#interface fa0/0
R1(config-if)#ip access-group ?
  <1-199>      IP access list (standard or extended)
  <1300-2699>  IP expanded access list (standard or extended)
  WORD         Access-list name

R1(config-if)#ip access-group 10 ?
  in   inbound packets
  out  outbound packets

R1(config-if)#ip access-group 10 out ?
  <cr>

R1(config-if)#ip access-group 10 out
R1(config-if)#
```

The extended ACL really lets you play with some power. This ACL is able to check the source and destination address, check source and destination TCP and UDP ports, or even check protocol type.

Once again, you may name your extended ACL, or simply number it. The acceptable numbers are 100 to 199 and 2000 to 2699.

Let us review the rather vast syntax options for the extended ACL. Notice in this example we are creating a named access control list:

```
R1(config)#ip access-list ?
  extended    Extended Access List
  log-update  Control access list log updates
  logging     Control access list logging
  resequence  Resequence Access List
  standard    Standard Access List

R1(config)#ip access-list extended ?
  <100-199>    Extended IP access-list number
  <2000-2699>  Extended IP access-list number (expanded range)
  WORD         Access-list name

R1(config)#ip access-list extended MYACL ?
  <cr>

R1(config)#ip access-list extended MYACL
R1(config-ext-nacl)#?
Ext Access List configuration commands:
  <1-2147483647>  Sequence Number
  default         Set a command to its defaults
  deny            Specify packets to reject
  dynamic         Specify a DYNAMIC list of PERMITs or DENYs
  evaluate        Evaluate an access list
  exit            Exit from access-list configuration mode
  no              Negate a command or set its defaults
  permit          Specify packets to forward
  remark          Access list entry comment

R1(config-ext-nacl)#permit ?
  <0-255>  An IP protocol number
  ahp      Authentication Header Protocol
  eigrp    Cisco's EIGRP routing protocol
  esp      Encapsulation Security Payload
  gre      Cisco's GRE tunneling
  icmp     Internet Control Message Protocol
  igmp     Internet Gateway Message Protocol
  ip       Any Internet Protocol
  ipinip   IP in IP tunneling
  nos      KA9Q NOS compatible IP over IP tunneling
  ospf     OSPF routing protocol
  pcp      Payload Compression Protocol
  pim      Protocol Independent Multicast
  tcp      Transmission Control Protocol
  udp      User Datagram Protocol

R1(config-ext-nacl)#permit tcp ?
  A.B.C.D  Source address
  any      Any source host
```

```
  host     A single source host

R1(config-ext-nacl)#permit tcp host ?
  Hostname or A.B.C.D  Source address

R1(config-ext-nacl)#permit tcp host 192.168.1.101 ?
  A.B.C.D  Destination address
  any      Any destination host
  eq       Match only packets on a given port number
  gt       Match only packets with a greater port number
  host     A single destination host
  lt       Match only packets with a lower port number
  neq      Match only packets not on a given port number
  range    Match only packets in the range of port numbers

R1(config-ext-nacl)#permit tcp host 192.168.1.101 any ?
  ack          Match on the ACK bit
  dscp         Match packets with given dscp value
  eq           Match only packets on a given port number
  established  Match established connections
  fin          Match on the FIN bit
  fragments    Check non-initial fragments
  gt           Match only packets with a greater port number
  log          Log matches against this entry
  log-input    Log matches against this entry, including input interface
  lt           Match only packets with a lower port number
  match-all    Match if all specified flags are present
  match-any    Match if any specified flag is present
  neq          Match only packets not on a given port number
  option       Match packets with given IP Options value
  precedence   Match packets with given precedence value
  psh          Match on the PSH bit
  range        Match only packets in the range of port numbers
  reflect      Create reflexive access list entry
  rst          Match on the RST bit
  syn          Match on the SYN bit
  time-range   Specify a time-range
  tos          Match packets with given TOS value
  ttl          Match packets with given TTL value
  urg          Match on the URG bit
  <cr>

R1(config-ext-nacl)#permit tcp host 192.168.1.101 any eq ?
  <0-65535>    Port number
  bgp          Border Gateway Protocol (179)
  chargen      Character generator (19)
  cmd          Remote commands (rcmd, 514)
  daytime      Daytime (13)
  discard      Discard (9)
  domain       Domain Name Service (53)
  drip         Dynamic Routing Information Protocol (3949)
  echo         Echo (7)
  exec         Exec (rsh, 512)
  finger       Finger (79)
  ftp          File Transfer Protocol (21)
```

```
  ftp-data      FTP data connections (20)
  gopher        Gopher (70)
  hostname      NIC hostname server (101)
  ident         Ident Protocol (113)
  irc           Internet Relay Chat (194)
  klogin        Kerberos login (543)
  kshell        Kerberos shell (544)
  login         Login (rlogin, 513)
  lpd           Printer service (515)
  nntp          Network News Transport Protocol (119)
  pim-auto-rp   PIM Auto-RP (496)
  pop2          Post Office Protocol v2 (109)
  pop3          Post Office Protocol v3 (110)
  smtp          Simple Mail Transport Protocol (25)
  sunrpc        Sun Remote Procedure Call (111)
  syslog        Syslog (514)
  tacacs        TAC Access Control System (49)
  talk          Talk (517)
  telnet        Telnet (23)
  time          Time (37)
  uucp          Unix-to-Unix Copy Program (540)
  whois         Nicname (43)
  www           World Wide Web (HTTP, 80)

R1(config-ext-nacl)#permit tcp host 192.168.1.101 any eq www
R1(config-ext-nacl)#
```

Since it is the same, there is no need to review the syntax of applying an extended access control list to an interface in a particular direction.

Note: An interface only permits one IPv4 ACL in one direction at a time. This is one of the reasons why it is critical to master the careful editing of these ACLs.

You can always view your access control lists on the device with **show access-list**. In order to confirm they are assigned to interfaces – use the command **show ip interface *inf_name***.

Wildcard Masking

Remember that a 0 in a bit position within an octet of the wildcard means to match the value of the corresponding address bit. As you might guess then, a 1 means to ignore the value of the corresponding address bit. This is why a wildcard mask of 0.0.0.0 indicates to match the entire 32 address bits exactly.

Let us say that we need to filter (deny) subnets 172.30.16.0/24 to 172.30.31.0/24.

The appropriate address and wildcard mask would be:

172.30.16.0 0.0.15.255

Notice how the wildcard mask matches the first two octets (172.30) of the IP address using all 0 bits in the first two octets of the wildcard mask. Because there is no interest in an individual host (thus a 0

value in the IP address forth octet), the wildcard mask ignores the final octet by using all 1 bits in the wildcard mask (resulting in a decimal value of 255). In the third octet, where the subnet address occurs, the wildcard mask of decimal 15, or binary 00001111, matches the high-order 4 bits of the IP address. In this case, the wildcard mask matches subnets starting with the 172.30.16.0/24 subnet. For the final (low-end) 4 bits in this octet, the wildcard mask indicates that the bits can be ignored. In these positions, the address value can be binary 0 or binary 1. Thus, the wildcard mask matches subnet 16, 17, 18, and up to subnet 31.

Remember that the keywords **host** and **any** can be used to represent wildcard masks of 0.0.0.0 and 255.255.255.255 respectively.

ACL Sequence Numbering

I remember vividly how thrilled I was when ACL sequence numbering finally arrived in the IOS. Notice that this takes place by default.

```
R1#show access-lists
Standard IP access list 10
    10 permit 192.168.1.0, wildcard bits 0.0.0.255 log
Extended IP access list 100
    10 permit tcp host 192.168.1.101 any eq www
```

You can use sequence numbers to insert statements anywhere in the named or numbered ACL. Benefits from these sequence numbers abound, including:

- Easy editing of a particular ACL entry
- The simple removal of a particular ACL entry
- The simple insertion of a particular ACL entry

Time-Based ACLs

As their name implies, time-based ACLs allows for access control that is based on the time of day and/or day of the week.

These powerful ACLs are actually very simple to configure. You create a time range that defines specific times of the day and week. You provide a name for the time range during its creation. This name is then referenced in the appropriate access control list.

Note: Remember that ACLs are used for much more than just filtering traffic. Therefore, your time-based ACL might also be useful in classifying traffic for some special QoS treatment, for example.

In this example, a Telnet connection is permitted from one network to the next on Monday, Wednesday, and Friday during business hours as defined in the time range.

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#time-range TR_EVERYOTHERDAY
R1(config-time-range)#periodic Monday Wednesday Friday 8:00 to 17:00
```

```
R1(config-time-range)#exit
R1(config)#access-list 100 permit tcp 192.168.1.0 0.0.0.255 172.16.1.0 0.0.0.255 eq
telnet time-range TR_EVERYOTHERDAY
R1(config)#
```

Note: Remember that the access list must be applied in order to be effective! Note also that the time range relies on the router system clock. It is critical, therefore, to ensure correct time on the device. We recommend the Network Time Protocol (NTP) for this function.

IPv6 ACLs

As you might expect, the ACL functionality in IPv6 is similar to that of ACLs in IPv4. The IPv6 access lists determine what traffic is blocked and what traffic is forwarded at router interfaces and allow filtering based on source and destination addresses, inbound and outbound to a specific interface. Just like with IPv4 ACLs, each list has an implicit deny statement at the end. IPv6 ACLs are defined and their conditions are set using the **ipv6 access-list** command with **deny** and **permit** keywords in global configuration mode. The command to assign the ACL to an interface is slightly different, the command is **ipv6 traffic-filter**. Notice the example below:

```
R1(config)#ipv6 access-list IPV6ACL
R1(config-ipv6-acl)#deny ipv6 fec0:0:0:2::/64 any
R1(config-ipv6-acl)#permit ipv6 any any
R1(config-ipv6-acl)#interface fa0/0
R1(config-if)#ipv6 traffic-filter IPV6ACL out
```

Common Issues with Access Control Lists

Here are the common issues that you should be acutely aware of regarding access control lists:

- The choice of access control list – remember that standard lists are perfectly appropriate when the only match criteria required is source address. With this said, analyze requirements very carefully. Often times a careful read of the requirements leads one to realize that an extended access control list is actually required since a specific protocol or application is needed in the match criteria.
- The order of ACL statements is absolutely critical – recall that the processing (testing) of the statements takes place in a top down direction. Statements that are more specific ***must*** go at the top of the list, otherwise the packet will be matched on a more general entry and the more specific match will never be tested.
- The implicit deny all – this means that every access control list must contain at least one permit statement.
- One access control list for IPv4 traffic, in one direction, per interface.
- Remember to carefully consider direction (inbound or outbound) when placing the list. Also, be sure to evaluate this direction with the source and destination logic inside the access control entries
- An ACL can filter traffic going through the router, or traffic to and from the router depending on how it is applied. Simple filtering ACLs cannot impact traffic generated by the router.

- If denying a particular host access to a resource using a standard ACL, consider placing this ACL as close to the resource (destination) as possible. Otherwise, the host will be denied from too many network resources.
- If denying a particular host access to a resource using an extended ACL, consider placing this ACL as close to the source as possible.
- When adding new ACL check for an existing ACL first. You do not want to overwrite an existing ACL accidentally.
- Do not apply an empty or non-existing ACL to the interface. It has no effect on the traffic and is a bad practice.
- Engineers fail to remark their ACLs or name them properly so they are unsure of their function.
- Test your ACLs before deployment. Engineers often forget this step. Also, consider using Notepad to assist in the creation and maintenance of the ACL.

Chapter Challenge: Access Control List Trouble Tickets

The following section includes sample trouble tickets for access lists. Remember to load the appropriate initial configurations for these sample tickets. Here is the simple topology that will be used for these tickets:

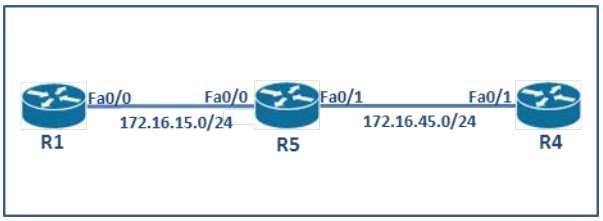


Figure 3-1 A Sample Troubleshooting Topology

Trouble Ticket #1

Ensure the ACL on R5 restricts Web requests from R1 Fa0/0 to Fa0/1 R4. Any other Web requests should be logged and include Layer 2 address information in the log message.

2 points

Trouble Ticket #2

Ensure the ACL on R4 logs incoming Telnet attempts but does nothing to block them or other traffic.

2 points

Chapter Challenge: Access Control List Trouble Ticket Solutions

The following section includes solutions to the Trouble Tickets associated with Access Control Lists.

Trouble Ticket #1

Ensure the ACL on R5 restricts Web requests from R1 Fa0/0 to Fa0/1 R4. Any other Web requests should be logged and include Layer 2 address information in the log message.

2 points

Step 1 - Fault Verification:

What is the current ACL policy on R5?

```
R5#show access-list
Extended IP access list FILTERWWW
    10 permit tcp any any
    20 permit tcp host 172.16.15.1 host 172.16.45.4 eq www
R5#show ip interface fa0/0
FastEthernet0/0 is up, line protocol is up
  Internet address is 172.16.15.5/24
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.10
  Outgoing access list is not set
  Inbound  access list is FILTERWWW
```

From this configuration we can see that an ACL named FILTERWWW has been created on R5 and assigned to the Fa0/0 interface inbound. The ACL has a problem, however. Line 10 will permit any Web requests to flow and is processed before line 20. Also, this access list will block any routing protocols in use. Also, the ACL does not log violations and include Layer 2 address information.

Step 2 – Fault Remediation

Here are the remediation steps required in this Trouble Ticket:

```
R5#show access-list
Extended IP access list FILTERWWW
    10 permit tcp any any
    20 permit tcp host 172.16.15.1 host 172.16.45.4 eq www
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#ip access-list extended FILTERWWW
R5(config-ext-nacl)#no 10
R5(config-ext-nacl)#30 permit eigrp any any
R5(config-ext-nacl)#40 deny ip any any log-input
R5(config-ext-nacl)#end
```

Notice that we remove line 10 and add lines 30 and 40 to meet the requirements.

Note: It would be very easy to forget to permit EIGRP traffic in this scenario. However, the **deny ip any any log-input** statement would actually make it easy to discover this after you apply. This is why this ACL entry is always a great idea in the lab exam, unless restricted by the instructions.

Step 3 – Fault Remediation Verification

Here we verify the ACL is correct:

```
R5#show access-list
Extended IP access list FILTERWWW
    20 permit tcp host 172.16.15.1 host 172.16.45.4 eq www
    30 permit eigrp any any (36 matches)
    40 deny ip any any log-input
```

Trouble Ticket #2

Ensure the ACL on R4 logs incoming Telnet attempts but does nothing to block them or other traffic.

2 points

Step 1 - Fault Verification:

What is the current ACL policy on R4?

```
R4#show access-list
Extended IP access list 100
    10 permit tcp any eq telnet any
R4#show ip interface fa0/1
FastEthernet0/1 is up, line protocol is up
  Internet address is 172.16.45.4/24
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.10
  Outgoing access list is not set
  Inbound  access list is not set
```

Notice that the access list logic is incorrect. In order to log Telnet attempts that are incoming, the destination port will be Telnet, not the source. In addition, this access list will block all other traffic due to the implicit deny all. Also, the Telnet line is missing the log keyword. Finally, this access list is not assigned!

Step 2 – Fault Remediation

Here is the process for correcting this Trouble Ticket:

```
R4#show access-list
Extended IP access list 100
    10 permit tcp any eq telnet any
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#ip access-list extended 100
R4(config-ext-nacl)#no 10
R4(config-ext-nacl)#permit tcp any any eq telnet log
R4(config-ext-nacl)#permit ip any any
R4(config-ext-nacl)#interface fa0/1
R4(config-if)#ip access-group 100 in
R4(config-if)#end
R4#
```

Notice how we can edit a numbered access list as if it were a named access list. This proves there is no real functional difference between the two.

Step 3 – Fault Remediation Verification

Here the verification involves a series of show commands as follows:

```
R4#show access-list
Extended IP access list 100
    10 permit tcp any any eq telnet log
    20 permit ip any any (48 matches)
R4#show ip interface fa0/1
FastEthernet0/1 is up, line protocol is up
  Internet address is 172.16.45.4/24
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.10
  Outgoing access list is not set
  Inbound  access list is 100
```

Chapter 4: Routing Protocol Security

The power of routers often times lies in their ability to run routing protocols. However, these routing protocols are far too willing to exchange information with unauthenticated sources. In this chapter, we examine how to ensure authentication in RIP version 2, EIGRP, OSPF, and BGP.

Routing Protocol Security Technology Review

RIPv2 and EIGRP Authentication

While often-made fun of equally, one must admit that RIP version 2 (RIPv2) is an improvement over its predecessor, RIP version 1 (RIPv1). While the major, revolutionary improvements were classless routing, RIPv2 also includes an option to authenticate updates. This authentication can be clear-text or MD-5 based. If you should choose MD5, the Cisco IOS Software uses the same key-chain mechanism that is used for EIGRP, this includes the ability to smoothly change the MD5 source key over time. As a result, this book will cover these RIPv2 and EIGRP authentication together.

RIPv2 Clear-Text Authentication

Before the MD5 method for EIGRP and RIPv2 are covered in this text, let us take a moment and examine a configuration that is very unrealistic for the real world, but that could appear in a certification environment. This configuration is clear-text authentication.

```
R1(config)#key chain KC_CLEARTEXT
R1(config-keychain)#key 1
R1(config-keychain-key)#key-string cisco
R1(config-keychain-key)#interface fa0/0
R1(config-if)#ip rip authentication mode text
R1(config-if)#ip rip authentication key-chain KC_CLEARTEXT
```

Is this configuration working? The great debug ip rip command can prove this. Notice in this example how updates are being ignored from a neighboring device that is not configured for authentication.

```
R1#debug ip rip
RIP protocol debugging is on
R1#
*Mar  1 00:10:00.523: RIP: sending v2 update to 224.0.0.9 via FastEthernet0/0
(172.16.15.1)
*Mar  1 00:10:00.523: RIP: build update entries - suppressing null update
*Mar  1 00:10:01.187: RIP: ignored v2 packet from 172.16.15.5 (invalid authentication)
R1#
```

RIPv2 and EIGRP MD5 Authentication

Now that the unrealistic case of RIPv2 clear text authentication is out of the way, let us examine MD5 authentication for RIPv2 and EIGRP.

As you witnessed for the RIPv2 clear text authentication example, a key element in the configuration of authentication for RIPv2 and EIGRP is the key chain.

Task 1: Create a key chain for RIPv2 and EIGRP that includes a long and random key string (password).

```
key chain KC_MYKEYS
key 1
key-string 46dgfGFT3646%^sdbi
```

You can optionally create additional keys with different send and accept lifetimes to enable smooth key rollover.

```
key2
key-string stvfi03w8eyII782rh
accept-lifetime 04:00:00 Jan 1 2014 04:00:00 Feb 1 2014 send-lifetime 04:00:00 Jan 1
2014 04:00:00 Jan 1 2014
```

Assigning these key chains to the authentication is simple:.

```
interface fa0/0
ip address 10.1.1.1 255.255.255.0
ip rip authentication mode md5
ip rip authentication key-chain KC_MYKEYS
ip authentication mode eigrp 100 md5
ip authentication key-chain eigrp 100 KC_MYKEYS
```

Note: The integer key IDs must match on all cooperating devices. Also note that if you do not use the **service password-encryption** command when you are implementing EIGRP authentication, the key string is stored as plaintext in the router configuration.

OSPF Authentication

OSPF authentication is the most complex of all of the routing protocols that we are responsible for. Remember this key point, OSPF authentication permits three different modes that can be configured in two different ways. Here are the modes of authentication that are available:

- Null – no authentication is performed on the link
- Clear Text – similar to the RIPv2 discussion, it is unrealistic to use this approach in the real world
- MD5 – this is the secure approach to use

The two methods of configuring one of these modes are as follows:

- Area-based
- Link-based

Clear Text OSPF Authentication

The first step to configure OSPF simple password authentication is to configure the key (password). This is accomplished using the **ip ospf authentication-key** interface configuration command. The password that is created by this command is used as a “key” that is inserted directly into the OSPF header when Cisco IOS Software originates routing protocol packets.

Note: A separate password can be assigned to each network on a per-interface basis and all neighboring routers on the same network must have the same password to be able to exchange OSPF information.

The second step to configure OSPF simple password authentication is to configure the authentication type. Specify the authentication type using the **ip ospf authentication** interface configuration command. Note that as described above, you can also set the authentication type for an area as well. If the authentication type is not specified for an interface, the authentication type for the area will be used.

The default authentication type for an OSPF area is Null of course. In order to enable authentication for an OSPF area, use the **area authentication** router configuration command.

Note: A very common problem when configuring OSPF authentication is the fact that an administrator will forget to authenticate a virtual link(s) used in the OSPF infrastructure. Remember that the virtual link is part of area 0, and may indeed need to be authenticated depending upon your configuration.

Examine this sample configuration of OSPF authentication on a link basis using the clear text approach:

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#int fa0/0
R5(config-if)#ip ospf authentication-key cisco123
R5(config-if)#ip ospf authentication ?
  message-digest  Use message-digest authentication
  null            Use no authentication
  <cr>

R5(config-if)#ip ospf authentication
R5(config-if)#end
R5#
*Mar  1 00:15:10.923: %SYS-5-CONFIG_I: Configured from console by console
R5#
*Mar  1 00:15:27.547: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.15.1 on FastEthernet0/0
from FULL to DOWN, Neighbor Down: Dead timer expired
R5#
```

Notice in the above example how the neighbor relationship between R5 and the neighboring R1 device is torn down as a result of the configuration. This is an expected result and confirms that the authentication configuration on the R5 device is done correctly. Of course, once the correct authentication configuration is performed on R1, the adjacency will re-establish.

Configuring MD5 Authentication for OSPF

The first step when configuring OSPF MD5 authentication is to configure the key ID and the key (password). Use the **ip ospf message-digest-key** command in order to complete this configuration. The key and the key ID that are specified in this command are used to generate a message digest (also called a hash) of each OSPF packet; the message digest is appended to the packet. A separate key can be assigned to each network on a per-interface basis. The key ID allows for a nice graceful transition in keys like we have with EIGRP. If an interface is configured with a new key, the router will send multiple copies of the same packet, each authenticated by different keys. The router will stop sending duplicate packets when it detects that all its neighbors have adopted the new key.

The second step when configuring OSPF MD5 authentication is to configure the authentication type. You can specify the authentication type using the **ip ospf authentication** command. For MD5 authentication, enter the **ip ospf authentication** command with the **message-digest** keyword. Remember that you can still set the MD5 authentication type for an area using the **area authentication** router configuration command. End

Here is an example of the MD5 router authentication configuration, accomplished in the area-based method:

```
R5(config)#interface fa0/0
R5(config-if)#ip ospf message-digest-key 1 md5 cisco123
R5(config-if)#router ospf 1
R5(config-router)#area 0 authentication message-digest
R5(config-router)#end
R5#
*Mar  1 00:25:56.263: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.15.1 on FastEthernet0/0
from FULL to DOWN, Neighbor Down: Dead timer expired
R5#
```

Once again, notice how the correct configuration tears down the adjacency.

BGP Routing Protocol Authentication

BGP neighbor authentication can be configured on a router so that the router authenticates the source of each routing update packet that it receives. This authentication is accomplished by the exchange of an authentication key (password) that is known to both the sending and the receiving router.

Unlike OSPF and RIPv2, BGP only supports MD5 neighbor authentication. To enable MD5 authentication on a TCP connection between two BGP peers, use the following command in BGP router configuration mode:

neighbor {ip-address | peer-group-name} password mypassowdstring

MD5 authentication must be configured with the same password on both BGP peers; otherwise, the connection between the peers will not be made.

Common Issues with Routing Protocol Security

Here are the most common issues involving authentication that we want you to be aware of:

- Mismatch in the authentication type
- Mismatch in the authentication key
- Mismatch in the authentication key ID
- Incorrect key rollover configuration

For examples of quick fire problem isolation and remediation, be sure to practice with the sample Trouble Tickets at the end of this chapter.

Chapter Challenge: Routing Protocol Security Trouble Tickets

The following section includes sample trouble tickets for routing protocol security. Remember to load the appropriate initial configurations for these sample tickets. Here is the simple topology that will be used for these tickets:

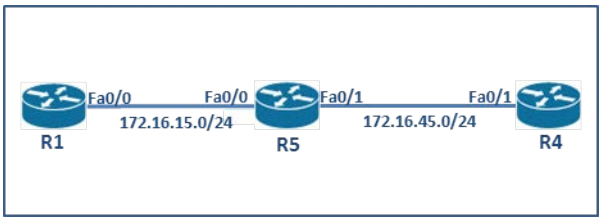


Figure 4-1 A Sample Troubleshooting Topology

Trouble Ticket #1

R5 is not learning the loopback 0 address of R1 via the dynamic routing protocol in use between R1 and R5. Discover and remediate the issue.

2 points

Trouble Ticket #2

R5 is not learning the loopback 0 address of R4 via the dynamic routing protocol in use between R4 and R5. Discover and remediate the issue.

2 points

Chapter Challenge: Routing Protocol Security Trouble Ticket Solutions

Trouble Ticket #1

R5 is not learning the loopback 0 address of R1 via the dynamic routing protocol in use between R1 and R5. Discover and remediate the issue.

2 points

Step 1 - Fault Verification:

What is the current routing table on R5?

```
R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.45.0 is directly connected, FastEthernet0/1
C       172.16.15.0 is directly connected, FastEthernet0/0
R5#
```

Notice that no remote networks are being dynamically learned. What protocol is in use between R5 and R1?

```
R5#show ip protocol
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 172.16.45.5
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    0.0.0.0 255.255.255.255 area 0
 Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.16.45.5          110      00:14:57
    172.16.15.1          110      00:14:57
  Distance: (default is 110)

Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
```

```
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 100
  EIGRP NSF-aware route hold timer is 240s
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.45.0/24
  Routing Information Sources:
    Gateway         Distance      Last Update
  Distance: internal 90 external 170

R5#
```

We can see from this output that it should be OSPF functioning between the two devices. Do we have R1 as an OSPF neighbor?

```
R5#show ip ospf neighbor

R5#
```

Is there an authentication configuration in place?

```
R5#show run | include authentication
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 KC_EIGRP
 area 0 authentication message-digest
R5#
```

Here we can see that MD5 authentication is intended on an area basis. Let us check the interface for its configuration:

```
R5#show run interface fa0/0
Building configuration...

Current configuration : 139 bytes
!
interface FastEthernet0/0
 ip address 172.16.15.5 255.255.255.0
 ip ospf message-digest-key 1 md5 cisco123
 duplex auto
 speed auto
end
```

Now let us move to R1 and check the authentication configuration:

```
R1#
R1#show run | section ospf
 ip ospf message-digest-key 2 md5 cisco123
router ospf 1
```

```
 log-adjacency-changes
 area 0 authentication message-digest
 network 0.0.0.0 255.255.255.255 area 0
R1#
```

We have apparently isolated the problem. There is a mismatch in the key ID.

Step 2 – Fault Remediation

Here is the process for correcting this Trouble Ticket:

```
R1#show run | section ospf
 ip ospf message-digest-key 2 md5 cisco123
router ospf 1
 log-adjacency-changes
 area 0 authentication message-digest
 network 0.0.0.0 255.255.255.255 area 0
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int fa0/0
R1(config-if)#no ip ospf message-digest-key 2 md5 cisco123
R1(config-if)#ip ospf message-digest-key 1 md5 cisco123
R1(config-if)#
*Mar  1 01:08:59.259: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.45.5 on FastEthernet0/0
from LOADING to FULL, Loading Done
R1(config-if)#
```

Notice that the adjacency has been reestablished. This is an excellent sign of course.

Step 3 – Fault Remediation Verification

Here the verification is simple:

```
R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/11] via 172.16.15.1, 00:00:55, FastEthernet0/0
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.45.0 is directly connected, FastEthernet0/1
C       172.16.15.0 is directly connected, FastEthernet0/0
R5#ping 1.1.1.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms
R5#
```

Trouble Ticket #2

R5 is not learning the loopback 0 address of R4 via the dynamic routing protocol in use between R4 and R5. Discover and remediate the issue.

2 points

Step 1 - Fault Verification:

What is the current routing table on R5?

```
R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/11] via 172.16.15.1, 00:03:54, FastEthernet0/0
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.45.0 is directly connected, FastEthernet0/1
C       172.16.15.0 is directly connected, FastEthernet0/0
R5#
```

Notice that no remote network is being dynamically learned. What protocol is in use between R5 and R1?

```
R5#show ip protocol
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 172.16.45.5
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    0.0.0.0 255.255.255.255 area 0
 Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.16.45.5          110      00:14:57
    172.16.15.1          110      00:14:57
  Distance: (default is 110)

Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
```

```
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 100
  EIGRP NSF-aware route hold timer is 240s
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.45.0/24
  Routing Information Sources:
    Gateway         Distance      Last Update
  Distance: internal 90 external 170

R5#
```

We can see from this output that it should be EIGRP functioning between the two devices. Do we have R4 as an EIGRP neighbor?

```
R5#show ip eigrp neighbor
IP-EIGRP neighbors for process 100
R5#
```

Is there an authentication configuration in place?

```
R5#show run | include authentication
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 KC_EIGRP
 area 0 authentication message-digest
R5#
```

Here we can see that MD5 authentication is intended for the peering. Let us check the key chain:

```
R5#show run | begin key chain
key chain KC_EIGRP
 key 1
   key-string cisco
```

Now let us move to R4 and check the authentication configuration:

```
R4#
R4#show run | include authentication
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 KC_EIGRP
R4#show run | begin key chain
key chain KC_EIGRP
 key 1
   key-string cisco
```

This configuration looks like it should work perfectly. A closer inspection at the CLI reveals that there is an extra space in the key string on R4.

Note: It is easy to discover extra spaces at the CLI by highlighting the key string and checking for extra characters.

Step 2 – Fault Remediation

Here is the process for correcting this Trouble Ticket:

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#key chain KC_EIGRP
R4(config-keychain)#key 1
R4(config-keychain-key)#key-string cisco
R4(config-keychain-key)#end
*Mar  1 01:22:02.015: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 172.16.45.5
(FastEthernet0/1) is up: new adjacency
R4(config-keychain-key)#end
R4#
```

Notice that the adjacency has been reestablished. This is an excellent sign of course.

Step 3 – Fault Remediation Verification

Here the verification is simple:

```
R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/11] via 172.16.15.1, 00:11:56, FastEthernet0/0
     4.0.0.0/24 is subnetted, 1 subnets
D       4.4.4.0 [90/409600] via 172.16.45.4, 00:00:09, FastEthernet0/1
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.45.0 is directly connected, FastEthernet0/1
C       172.16.15.0 is directly connected, FastEthernet0/0
R5#ping 4.4.4.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/21/24 ms
```

R5#

Chapter 5: Catalyst Security

You cannot be one of those network administrators that forgets to secure your Layer 2 of the OSI model. Securing Layer 2 is so valuable because it ends up impacting the security of the layers above them! This chapter walks you through some remarkable security options that are available on the Cisco Catalyst switches.

Catalyst Security Technology Review

Before we focus on what can go terribly wrong with our Catalyst security, and enjoy some sample Trouble Tickets, let us review topic by topic that we want to focus on in this area.

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast "storm" on one of the physical interfaces. We consider a storm a flood of packets (often caused by a malicious attack or errors in hardware and/or software) that can disrupt the normal flow of the traffic we need to move through our network.

Storm control (sometimes referred to as traffic suppression in the literature) monitors packets and first determines if the packet is unicast, multicast, or broadcast in form. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

When you configure storm control, you can use one of several methods for having the switch quantify the actual storm condition:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in packets per second for small frames

No matter which method of traffic measurement you select, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level.

Note: When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, is blocked. Be aware, however, that routing protocol traffic (such as OSPF) cannot be classified as control traffic and would be dropped.

The interface configuration command for storm control is as follows:

storm-control {**broadcast** |**multicast** | **unicast**} **level** {*level*[*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}

Optionally, you can also specify particular actions to be taken (other than the default blocking behavior) when the storm occurs. This is done with the following interface configuration command:

storm-control action {**shutdown** |**trap**}

The **shutdown** keyword causes the port to error-disable during a storm, while the **trap** keyword generates an SNMP trap when a storm is detected.

Small Frames

Incoming VLAN-tagged packets smaller than 67 bytes are considered *small frames* by the Cisco IOS. They are forwarded by the switch, but they do not cause the switch storm-control counters to increment. In Cisco IOS Release 12.2(44)SE and later, you can configure a port to be error disabled if small frames arrive at a specified rate (threshold).

You must globally enable the small-frame arrival feature on the switch and then configure the small-frame threshold for packets on each interface. Packets smaller than the minimum size and arriving at a specified rate (the threshold) trigger an error-disabling of the port.

If the **errdisable recovery cause small-frame** global configuration command is entered, the port is re-enabled after a specified time. Use the **errdisable recovery** global configuration command in order to configure the rate. Here is an example of this configuration:

```
SW1# configure terminal
SW1(config)# errdisable detect cause small-frame
SW1(config)# errdisable recovery cause small-frame
SW1(config)# interface fa0/1
SW1(config-if)# small-frame violation rate 10000
SW1(config-if)# end
```

Common Issues with Storm Control

Please carefully review these common issues when dealing with the storm control feature in a production and/or certification environment:

- As always, be sure you are applying the feature on the correct interface for the correct device
- Ensure you are setting your rising and falling threshold correctly per the scenario's requirements
- Ensure that you are configuring the correct measurement system – for example, per packet or per bits per second

Port Security

Easily one of the most powerful security mechanisms available at Layer 2 is the Port Security option. This simple and easy to understand feature allows you to limit and even preconfigure the MAC addresses permitted to communicate through a port. The MAC addresses can be learned dynamically, statically configured by the administrator, or dynamically learned then statically written by the IOS in a process called sticky learning. Let us examine each of these methods in greater detail:

- Dynamic port security – with this configuration, you specify how many different MAC addresses are permitted to use a port at one time. You might be thinking, "Why would there ever be more than one in full duplex LAN environment?" Let us not forget Voice over IP (VoIP) environments. Notice that you use the dynamic approach when the concern is merely the number of IP

addresses as opposed to the specific addresses that are permitted. Depending on how you configure the switch, these dynamically learned addresses age out after a certain period, and new addresses are learned, up to the maximum that you have defined.

- Static port security – with this configuration, you statically configure which specific MAC addresses are permitted to use a port. Any source MAC addresses that you do not specifically permit are not allowed to source frames to the port.
- A combination of static and dynamic port security – with this configuration, you can choose to specify some of the permitted MAC addresses and let the switch learn the rest of the permitted MAC addresses. For example, if the number of MAC addresses is limited to five, you could statically configure two MAC addresses and the switch dynamically learns the next three MAC addresses that it receives on that port. Port access is limited to these five addresses. Note that the two statically configured addresses do not age out, but the three dynamically learned addresses can, depending on the switch configuration.
- Sticky learning port security – with this configuration, the interface dynamically learns the MAC addresses on the interface up to the configured (or default) limit and writes these entries as static entries in the running configuration. Then, all that is left for the administrator to do is save the running configuration to the startup configuration in order to have a static port security environment. Note that sticky learned MAC addresses do not age out.

Remember that there are three options when it comes to the response that a switch can take to a violation in the port security feature. These are aptly termed the violation modes:

- Shutdown – this is the default violation mode and also the most strict. In this mode, the port is dynamically error disabled. Note that this is the strictest response possible to a violation.
- Protect – this mode disallows the MAC address in violation from communicating, put keeps the port enabled for other permitted MAC addresses to use. This mode does not log the violation occurred in any way, and as a result, this mode is not recommended for use by Cisco. Why they even give us this option is a bit of a mystery.
- Restrict – this mode is just like the Protect mode, except it does log the violation has occurred.

Here is a sample dynamic port security configuration:

```
SW1(config)# interface fa0/1
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan 10
SW1(config-if)# switchport port-security maximum 2
SW1(config-if)# switchport port-security
```

Note: The port needs to be in access mode in order for the feature to be configured. Also, notice how it could be fairly easy to forget to actually enable the feature with the **switchport port-security** command. Finally, in static port security configurations, remember to accommodate potential virtual MAC addresses in your topology.

Common Issues with Port Security

Please carefully review these common issues when dealing with the port security feature in a production and/or certification environment:

- As always, be sure you are applying the feature on the correct interface for the correct device
- Ensure you set the violation mode per the scenario; remember that while the protect mode is never recommended by Cisco for a production environment, in the certification exam environment, anything goes
- Watch out for "nonstandard" MAC addresses that might exist in your scenario, a famous example being a virtual MAC address for a technology like HSRP

802.1X

Cisco's Identity Based Networking Services (IBNS) is an integrated solution combining several Cisco products that offer authentication, access control, and user policies to secure network connectivity and resources. Here, we describe an important aspect of Cisco IBNS that is on the blueprint for the CCIE R&S - IEEE 802.1X.

Cisco IBNS is a solution for increasing the security of physical and logical access to an enterprise network that is built on the IEEE 802.1X standard. Cisco IBNS allows you to implement true identity-based network access control and policy enforcement at the user and port levels. It provides user and device identification using secure, reliable, and strong authentication technologies. The Cisco IBNS solution associates identified entities with policies that are created and administered by management and provides increased granularity of control. The Cisco IBNS solution is based on standards-based RADIUS and 802.1X implementations. It interoperates with all Internet Engineering Task Force (IETF) authentication servers that comply with these two standards. Cisco has enhanced its Cisco Secure Access Control Server (ACS) to provide a tight integration across all Cisco switches.

These capabilities are introduced when a Cisco end-to-end system is implemented with the Cisco Catalyst family of switches, wireless LAN (WLAN) access points (APs) and controllers, and Cisco Secure ACS. Additional components of the system include an 802.1X-compliant client operating system, such as Microsoft Windows 7 or Apple Mac OS X, and an optional X.509 public key infrastructure (PKI) certificate architecture. Cisco IP phones also interoperate with an identity-based networking system that is based on 802.1X, when it is deployed on a Cisco end-to-end infrastructure.

IEEE 802.1X is a standard set by the IEEE 802.1 working group and is a framework designed to address and provide port-based access control using authentication. 802.1X authenticates network clients using information unique to the client and with credentials known only to the client. This service is called port-level authentication because it is offered to a single endpoint for a given physical port. The 802.1X standard defines a client- and server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly

authenticated. The authentication server authenticates each client that is connected to a switch port before making available any services that the switch or LAN offers.

Until the client is authenticated, 802.1X access control allows only EAP over LAN (EAPOL), Cisco Discovery Protocol, and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Network devices (switches or APs) perform the enforcement of 802.1X controlled access through MAC-address-based filtering, allowing only authenticated MAC addresses access though a switch port. Through port state monitoring, the authentication process restarts every time an interface (port) becomes active.

With 802.1X port-based authentication, devices in the network have specific roles: the supplicant, the authenticator, or the authentication server. The supplicant is a device (workstation, laptop, and so on) that requests access to the LAN and switch services, and responds to requests from the authenticator (switch). The device must run 802.1X-compliant client supplicant software, such as Cisco Secure Services Client. Cisco also offers an 802.1X supplicant on Cisco IP phones. The supplicant and the authenticator communicate exclusively using the Extensible Authentication Protocol over LAN (EAPOL). The authenticator is a device, such as a Cisco Catalyst switch, that controls access to the network based on the authentication status of the client. The authenticator usually acts as an intermediary (proxy) between the client and the authentication server. The authenticator requests identity information from the client, forwards this information to the authentication server, and then relays the response of the authentication server to the client.

The authentication server performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication server is transparent to the client. The authenticator and the authentication server communicate exclusively over the RADIUS protocol.

In compliance with the 802.1X standard, authenticators perform basic port-based authentication and network access control. When the 802.1X-compliant client software is configured on the end device (client), a Cisco Catalyst switch running 802.1X authenticates the requesting user or system with a back-end Cisco Secure ACS or other RADIUS server.

Here is how 802.1X port-based access control works within Cisco IBNS.

Step 1 - A client connects to an 802.1X-enabled network and sends a start message to the LAN switch using EAPOL.

Step 2 - The LAN switch sends a login request to the client.

Step 3 - The client replies with a login response.

Step 4 - The switch forwards the response to the authentication server.

Step 5 - The authentication server authenticates the user.

Step 6 - The authentication server authorizes network access for the user and informs the LAN switch over RADIUS.

Step 7 - The LAN switch then enables the port that is connected to the client and applies optional authorization parameters.

The actual authentication conversation occurs between the client and the authentication server using EAP. The authenticator is aware of this activity but is just an intermediary. 802.1X uses the Extensible Authentication Protocol (EAP) to authenticate users who wish to access the network. EAP messages are exchanged between a supplicant and an authenticator, which are tunneled inside the EAPOL and RADIUS protocols. The authenticator is aware of this interaction, but does not actively participate in it. Instead, it waits for the decision of the authentication server, which is communicated to it natively over the RADIUS protocol. The specific exchange of EAP frames depends on the authentication method (that is, EAP variety)being used.

Using this architecture, 802.1X not only provides the capability to permit or deny network connectivity that is based on user or machine identity, but also works with higher-layer protocols to enforce network policy.

Cisco Catalyst IOS Software includes a rich 802.1X authenticator function that includes the following:

- Standard 802.1X/EAPOL implementation
- RADIUS support
- Flexible handling of exceptions (guests, non-802.1X-capable devices)
- Additional authorization options (per-user VLAN/ACL assignment)
- Ease-of-deployment features

The Cisco Catalyst IOS Software includes a rich and flexible 802.1X authenticator function that allows highly customizable integration with enterprise supplicant and authentication server environments. The software provides a standard 802.1X function using EAPOL exchanges over switch LAN ports and a RADIUS client to integrate with RADIUS authentication servers. The 802.1X standard does not specify all aspects of an 802.1X-enabled network. For example, the standard does not mandate what the authenticators should do with unresponsive supplicants, or supplicants failing authentication, beyond not allowing these supplicants regular network access.

The Cisco Catalyst IOS Software provides users with many additional options that can provide simple guest access and create centralized exception policies that will allow limited access to non-802.1X-compliant devices. In addition to flexible exception handling, Cisco Catalyst IOS Software allows for the creation of powerful per-user or per-group authorization rules, allowing flexible access control through dynamic VLAN or ACL assignment.

Finally, the Cisco Catalyst IOS Software supplicant includes many ease-of-use features. These features include 802.1X critical ports, 802.1X open authentication, and 802.1X unidirectionally controlled ports,

which allow for smoother deployment of 802.1X functionality in many enterprise environments and ensure compatibility with most existing network endpoints.

When 802.1X is enabled on an access port, a user without an 802.1X client is typically denied access to the network. The guest VLAN feature of 802.1X offers limited network access to these users. You can configure a guest VLAN for each 802.1X port on the switch. Users may not have an 802.1X client, because they are upgrading their system or they are using a system that is not 802.1X-capable, such as Microsoft Windows 98.

If a guest VLAN is enabled on an 802.1X port, the switch assigns a client to the guest VLAN if the client does not send EAPOL packets or does not reply to the EAP Request/Identity frame from the authentication server. Any number of 802.1X-incapable clients are allowed access when the switch port is moved to the guest VLAN.

Note: If an 802.1X-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted

You can configure a restricted VLAN (sometimes called an authentication-failed VLAN) for each 802.1X port on a switch to provide limited services to clients who cannot access the guest VLAN. These clients are 802.1X-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users who do not have valid credentials on an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.

Note: A VLAN can be configured to be both the guest VLAN and the restricted VLAN, if you want to provide the same services to both types of users. The switch (authenticator) keeps a count of failed authentication attempts for any client. When the count exceeds the maximum number of authentication attempts that are configured, the port moves to the restricted VLAN. When the RADIUS server replies with either an EAP Failure or an empty response that does not contain an EAP packet, the failed attempt count is incremented. The failed attempt counter resets when the port moves into the restricted VLAN.

Cisco Secure ACS can perform per-user and per-group authorization for network access. This is accomplished through the following:

- VLAN assignment: To assign a specific VLAN for a user or a group
- ACL assignment: To assign a specific port ACL (PACL) for a user or a group
- Time-based access: To limit access to the network based on time of day
- MAC address bypass accounts: As one possible method to handle 802.1X-exceptions

In addition to acting as an authentication server in IBNS solutions, which provides the functions of an EAP server, Cisco Secure ACS products can provide powerful, scalable, and easy-to-use per-user and per-group authorization features. These features include the following:

- VLAN assignment, in which the Cisco Secure ACS authentication server can associate a VLAN with a particular user or group, and instruct the switch to dynamically assign the authenticated user into that VLAN. If your organization uses an access control method between VLANs (for example, routed ACLs or a firewall system), this can easily provide strong access control and auditing in the internal network.
- Access control list (ACL) assignment, in which the Cisco Secure ACS authentication server can associate a stateless ACL with a particular user or group, and instruct the switch to dynamically assign the ACL to the port of the user. This mechanism can provide a very granular access control method because it extends to the port level, but it is not able to support all network applications because stateful packet filtering at the switch port level is lacking.
- Time-based access, in which the Cisco Secure ACS authentication server can only allow users to log into the network during specific periods.
- MAC address bypass authorization, as one of possible methods to create exceptions for non-802.1X-enabled hosts

When you are preparing your 802.1X deployment, you should first gather a list of LAN switches potentially allowing unauthorized access. You should also identify the available user authentication credentials. Finally, you should identify the types of clients that will be requiring access to your infrastructure. You might also want to examine automated software deployment options for the client software.

The basic 802.1X deployment steps consist of the following (you should note that not all steps are relevant in the CCIE R&S environment but are listed here for completions sake):

1. Configure core switch 802.1X authenticator features.

2. (Optional) Configure additional switch 802.1X authenticator features (guest/restricted VLAN, reauthentication, timers).

3. Configure Cisco Secure ACS for EAP-FAST.

4. Configure and deploy the Cisco Secure Services Client wired 802.1X supplicant.

Let us now examine the configuration and verification of the Catalyst switch as the 802.1X authenticator. I know, I know, you are probably thinking that it is about time. Here is the process we need to be intimately aware of:

1. Configure a RADIUS association between the switch and the AAA server.

2. Enable AAA and use RADIUS for network AAA authentication.

```
aaa new-model
aaa authentication dot1x default group radius
!
radius-server host 10.1.1.1 key cisco123
```

3. Enable 802.1X globally on a switch.

```
!
dot1x system-auth-control
!
```

4. Enable 802.1X port control on user switch ports.

```
interface range FastEthernet0/1 - 24
switchport mode access
switchport access vlan 10
authentication port-control auto
```

5. (Optional) Configure periodic reauthentication.

6. (Optional) Tune timers and thresholds.

7. (Optional) Configure a guest access policy on the switch.

You should notice that many of these configuration steps are indeed optional. Also, you should note that there are many steps here that need to be configured. Knowing exactly where this information is located in the Cisco Documentation can be very beneficial as this involved configuration might be forgotten.

Note: It is generally recommended that you configure at least two RADIUS servers for redundancy and to provide appropriate path redundancy to both RADIUS servers.

As you know – verification is of critical importance with configurations such as these. One of my favorite verifications here is to verify the detailed 802.1X status of a port:

```
Switch#show dot1x interface Fa0/1 details
Dot1x Info for FastEthernet0/1
-----------------------------------
PAE = AUTHENTICATOR
PortControl = FORCE_AUTHORIZED
ControlDirection = Both
HostMode = SINGLE_HOST
Violation Mode = PROTECT
ReAuthentication = Enabled
QuietPeriod = 5
ServerTimeout = 30
SuppTimeout = 30
ReAuthPeriod = 600 (Locally configured)
...
Auth-Fail-Vlan = 700
Auth-Fail-Max-attempts = 1
Guest-Vlan = 700
Dot1x Authenticator Client List Empty
Domain = ALL
Port Status = AUTHORIZED
```

Notice that the port status is AUTHORIZED. This is our ultimate verification! You can also see that I configured optional elements on the device. Since you are not responsible for those configurations at this point, I did not show them in our earlier text.

Common Issues with 802.1X

There are indeed common issues that can occur with your 802.1X environment. Let us examine those and the steps that you should take in response:

- You discover that the supplicant on your network cannot communicate properly on your network - first verify that your 802.1X supplicant is configured properly. You can use the **dot1x test eapol-capable** command to verify that the supplicant responds to EAPOL requests. If it does not, verify your configuration using the Cisco Secure Services Client client GUI or its configuration Management Utility. If the supplicant responds to EAPOL requests, proceed to the next step.
- Verify that the RADIUS and EAP associations are properly set up. You can determine this by observing the switch logging output. This output will indicate RADIUS issues by using the switch CLI **test aaa** command to make test AAA authentication transactions against the RADIUS servers and by observing the Cisco Secure ACS Failed Attempts report to observe issues as seen by the ACS. If there does not appear to be problems with RADIUS or EAP, proceed to the next step.
- Verify the authentication process on the Cisco Secure ACS by observing its Failed Attempts report, and look for possible user credential problems, such as bad passwords or failed external databases.

VLAN Access Maps

We know that Router-Based Access Control Lists or RACLs can be used to control packets flowing between VLANs at a Cisco route processor. However, what about the ability to control traffic *within* a VLAN. This is the job of VLAN-Based Access Control Lists or VACLs. Note that because of the command syntax used to create them, VACLs are often referred to as VLAN Access Maps. This is excellent nomenclature since they are also very similar to Route Maps in their construction.

So be sure to consider VLAN Access Maps when security packet filtering is required within a VLAN. Note that because these maps control the intra-VLAN traffic, they are not defined by direction (input or output).

VLAN Access Maps provide a wide variety of criteria for matching traffic. Of most importance is your ability to configure VLAN Access Maps to match on Layer 3 addresses (IP addresses) for IPv4 traffic.

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. Keep in mind that you can enforce VLAN Access Maps only on packets going through the switch; you cannot enforce VLAN Access Maps on traffic between hosts on a hub or on another switch connected to your switch. Also, while there are many different criteria you can specify in your VLAN Access Maps, only one is permitted per VLAN.

If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet should the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Step 1 – Create a standard or extended IPv4 ACL or a named MAC extended ACL that you intend to apply to the VLAN.

Step 2 – Use the **vlan access-map** global configuration command to create a VLAN ACL map entry.

Step 3 – In the VLAN access-map configuration mode, use the **match** command and **forward** or **drop** actions.

Step 4 – Use the **vlan filter** global configuration command to apply the VLAN map to one of more VLANs.

Keep in mind when creating your VLAN Access Maps that the order of entries in your maps is critically important. Just like with a Router-based Access Control List, a packet that comes into the switch is tested against the first entry in the VLAN map. If there is a match, the action specified for that part of the VLAN map is followed. If there is no match, the packet is tested against the next entry in the map.

It is simple to delete VLAN Access Maps. Just use the **no vlan access-map** global configuration command. You can specify the VLAN Access Map name and sequence number should you just need to delete a particular entry from within the map.

Now let us look at an example.

```
Switch(config)# ip access-list extended AL_SAMPLE
Switch(config-ext-nacl)# permit tcp host 10.10.10.1 host 10.10.10.2
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map VACL_SAMPLE 10
Switch(config-access-map)# match ip address AL_SAMPLE
Switch(config-access-map)# action drop
Switch(config)# vlan access-map VACL_SAMPLE 20
Switch(config-access-map)# action forward
```

Notice that we first create an Access List named AL_SAMPLE that permits all TCP-based traffic going from a certain host to another host. We then create a VLAN Access Map named VACL_SAMPLE and enter a sequence number of 10. We match on the Access List we created (AL_SAMPLE) and drop traffic matching this access control list. Notice the next VLAN Access Map entry that matches on all remaining traffic (there is no criteria specified) and specifies a forward action.

Always remember that the VLAN Access Map must be applied to actually take effect! This can be done as follows:

```
Switch(config)# vlan filter map VACL_SAMPLE vlan-list 20-22
```

Notice here we are applying our VLAN Access Map to the VLANs 20 through 22.

Common Issues with VLAN Access Maps

As you might guess, there are several common issues to watch out for when working with VLAN Access Maps. These issues include:

- Configuration of the VLAN Access Map on the incorrect device. Remember, the VLAN Access Map needs to be placed on the device where traffic is expected to move through.
- Failure to apply the VLAN Access Map to the correct VLAN.
- Incorrect order of entries in the VLAN Map itself.
- Failure to consider and accommodate the default drop actions of the VLAN Access Map.
- Incorrectly constructed match criteria. Typically, this is an improperly constructed Access Control List or MAC Access List.

DHCP Snooping

DHCP Snooping is a powerful Catalyst security mechanism that can help guard against a variety of security issues surrounding the important and common Dynamic Host Configuration Protocol (DHCP).

One of the most important areas it guards against is the rogue DHCP server. A rogue DHCP server is one that you did not authorize and that was created accidentally or maliciously in our network. Here is the simple process by which this rogue device can create problems:

1. An attacker activates a malicious rogue DHCP server on the port of a switch in the network infrastructure.

2. One of your network clients broadcasts a DHCP configuration request.

3. The DHCP server of the attacker responds before the legitimate DHCP server can respond, assigning IP address information that the attacker wants the client to possess. Perhaps this is the first step of a later attack, or the attacker simply wants to provide IP configuration information that will prevent the client from communicating on the network. The attacker might even provide its own address to the client as the DHCP-assigned default gateway information. Now the client is directing packets to the attacker that it would otherwise send to a legitimate default gateway.

Another attack type to consider that is very easy to implement is a DHCP starvation attack. This attack works by broadcasting DHCP requests with spoofed MAC addresses. If enough requests are sent, the network attacker can exhaust the address space available on the legitimate DHCP servers for some period of time. While this is successful enough in itself to cause a network disruption, the attacker can now set up a rogue DHCP server and respond to new DHCP requests from clients on the network.

What would be a simple protection mechanism against this DHCP starvation attack? Well, we already presented it in this chapter – the powerful port security feature. This feature would prevent the spoofed MAC address initial and falsified DHCP requests.

However, what about the rogue DHCP server itself that we have mentioned in two types of attacks now? This is where DHCP snooping can be so valuable. With DHCP snooping, the administrator designates switch ports as trusted or untrusted. By default, all ports in the network infrastructure are untrusted.

The ports the administrator designates as trusted ports can forward DHCP requests and acknowledgements. The ports that remain as untrusted ports can forward only DHCP requests.

DHCP snooping enables the switch to build a table that maps a client MAC address, IP address, VLAN, and port ID. As you will learn later in this chapter, this important table can be leveraged to provide other security controls as well.

Remember, for DHCP snooping to work, each switch port must be labeled as trusted or untrusted. Trusted ports are the ports over which the DHCP server is reachable and that will accept DHCP server replies. All other ports should be labeled as untrusted ports (and they are by default) and can only source DHCP requests. Typically, this means the following:

- All access ports should be labeled as untrusted, except the port to which the DHCP server is directly connected
- All inter-switch trunk ports should be labeled as trusted
- All ports over which the reply from the DHCP server is expected should be labeled as trusted
- Untrusted ports are those ports that are not explicitly configured as trusted

Here are the steps to configure DHCP snooping:

Step 1. Enable DHCP snooping globally on a switch.

```
ip dhcp snooping
```

Step 2. (Optional) Specify the location of the persistent DHCP snooping binding database.

```
ip dhcp snooping database flash:/dhcp-snooping.db
```

Step 3. Designate ports that forward traffic toward the DHCP server as trusted.

```
interface FastEthernet 5/1
ip dhcp snooping trust
```

Step 4. Designate all other ports (including those with statically addressed hosts) as untrusted.

Step 5. (Optional) Configure DHCP rate limiting (and port security) on each untrusted port.

```
interface range FastEthernet 5/3 – 24
ip dhcp limit rate 2
```

Step 6. Enable DHCP snooping in specific VLANs.

```
ip dhcp snooping vlan 500
```

Your Cisco IOS Software includes rich DHCP snooping validation features. Use the **show ip dhcp snooping** command to display the general DHCP snooping configuration for a switch. You can also use the **show ip dhcp snooping binding** command to display the dynamically discovered bindings in the DHCP snooping binding database. To filter which addresses are displayed, provide either a MAC address or an IP address for the address parameter.

The **show ip dhcp snooping database** command displays the status of the binding database. This command is especially useful when using a remote storage server to determine its current state and historical operational statistics.

Common Issues with DHCP Snooping

With DHCP snooping configurations, be sure to troubleshoot the following common issues:

- Ensure the security feature is enabled globally
- Ensure the appropriate ports are trusted – these include the access port connected to the legitimate DHCP server and the inter-switch ports in the topology
- Ensure the DHCP snooping feature is enabled for the correct VLANs

Dynamic ARP Inspection

Remember earlier that we stated DHCP Snooping could lead to other excellent Catalyst security mechanisms. One of those is DAI or Dynamic ARP Inspection. You recall that in normal Address Resolution Protocol (ARP) operation, a host sends a broadcast to determine the MAC address of a destination host with a particular IP address. The device with the IP address replies with its MAC address. The originating host caches the ARP response, using it to populate the destination Layer 2 header of packets that are sent to that IP address. By spoofing an ARP reply from a legitimate device with a gratuitous ARP (GARP), an attacking device can appear to be the destination host that the sender is seeking.

The ARP reply from the attacker causes the sender to store the MAC address of the attacking system in its ARP cache. As you might guess, all packets that are destined for that IP address are forwarded to the attacker system.

To address this ARP vulnerability in the infrastructure, you can use Static or Dynamic ARP Inspection in your network switches, or use static ARP entries on your infrastructure devices, therefore, eliminating ARP on critical areas of your network.

With ARP inspection, you once again designate ports as trusted or untrusted. Trusted ports are allowed to forward ARP messages while ARP messages on untrusted ports undergo ARP inspection validation. Your Catalyst switch can perform the ARP validation against a static ARP ACL for static IP addresses (we call this static ARP inspection) or against the DHCP snooping binding database for your DHCP assigned IP addresses (we call this Dynamic ARP Inspection).

In general, you should consider following these basic guidelines with this feature:

- Leave ports as untrusted that connect to any host that is considered a possible source of attack.

- For hosts with static IP addresses, use static ARP ACL entries on the switch to permit their ARP traffic.
- Ensure you configure all ports to other switches that do not support ARP inspection as untrusted.
- Configure all other ports as trusted.

Follow these steps in order to configure ARP Inspection:

Step 1. (Optional) If using DHCP, verify that DHCP snooping is active and has fully populated its databases.

Step 2. (Optional) Designate ports on which the risk of ARP spoofing is acceptable as trusted.

```
interface fa0/10
 ip arp inspection trust
```

Step 3. Designate all other ports as untrusted.

Step 4. (Optional) Tune the ARP rate limit on each port.

```
interface fa0/8
 ip arp inspection limit rate 50
```

Step 5. (Optional) Configure an ARP ACL with static IP-MAC mappings.

```
arp access-list MYARPEXCEPTION
 permit ip host 192.168.1.1 mac host 00d1.0cc9.01b8
!
ip arp inspection filter MYARPEXCEPTION vlan 500
```

Step 6. (Optional) Tune the error-disable behavior.

```
errdisable recovery cause arp-inspection interval interval
```

Step 7. Enable ARP snooping in a specific VLAN

```
ip arp inspection vlan 500
```

When you are ready to verify your configuration, you can use the **show ip arp inspection** command to display the general ARP inspection configuration for your switch.

Note: There are indeed several additional validation (protocol verification) checks that ARP inspection can apply to ARP traffic, and these are all disabled by default. Be careful when implementing this feature since some clients may use nonstandard ARP messages that are dropped by these checks.

Another powerful verification is to use **show ip arp inspection interfaces** this is a quick way to show the trust mode of interfaces and their settings.

Common Issues with DAI

Be aware of the most common issues with DAI:

- The feature is not configured on the appropriate VLAN
- The correct ports have not been configured for trusted/untrusted status
- Static entries have not been properly accommodated
- The DHCP Snooping database is non-existent or stale

IP Source Guard

Yet another feature that DHCP Snooping makes possible is the IP Source Guard security feature. IP Source Guard is a Cisco IOS Software feature that can stop an IP spoofing attack closest to the source—on the switch port where the attacking host connects to the network. Like DHCP snooping, you can enable IP Source Guard on a DHCP snooping untrusted port.

Initially, all IP traffic on the port is blocked except for DHCP packets that are captured by the DHCP snooping process. When a client receives a valid IP address from the DHCP server, or when the user configures a static IP source binding, IP Source Guard installs an autogenerated Port-Based ACL (PACL) on the interface. The PACL allows only IP traffic from a source IP address that is in the IP source binding table and denies all other traffic. The PACL takes precedence over any router ACLs or VLAN maps that affect the same interface.

Note: Turn on DHCP snooping at least 24 hours before enabling IP Source Guard for the switches to learn all IP addresses. Alternatively, you can force DHCP renewals using some other method to accelerate deployment.

You can also deploy IP Source Guard on ports that connect statically addressed systems to the network by specifying static mapping in the IP Source Guard configuration. Alternatively, you can use manually configured PACLs on the same port.

IP Source Guard supports only Layer 2 ports, including both access ports and trunk ports. For each untrusted Layer 2 port, there are two levels of IP traffic security filtering:

- Source IP address filter: IP traffic is filtered based on its source IP address. Only IP traffic with a source IP address that matches the IP source binding entry is permitted. An IP source address filter is changed when a new IP source entry binding is created or deleted on the port. The PACL

is recalculated and reapplied in the hardware to reflect the IP source binding change. By default, if the IP filter is enabled without any IP source binding on the port, a default PACL that denies all IP traffic is applied to the port. Similarly, when the IP filter is disabled, any IP source filter PACL is removed from the interface.

- Source IP and MAC address filter: IP traffic is filtered based on its source IP address and its MAC address. Only IP traffic with source IP and MAC addresses that match the IP source binding entry are permitted.

Note: If you enable IP Source Guard on a trunk port with many VLANs that have DHCP snooping enabled, you could run out of ACL hardware resources and some packets might be switched in software.

Follow these steps in order to configure IP Source Address control:

Step 1. (Optional) If using DHCP, verify that DHCP snooping is active and has fully populated its databases.

Step 2. Enable IP Source Guard on DHCP snooping enabled ports.

```
interface FastEthernet 5/1
 ip verify source port-security
```

Step 3. (Optional) Configure a static IP Source Guard mapping or port ACLs (PACLs) on ports connecting to statically addressed hosts.

```
ip source binding 00:01:00:A0:04:29 vlan 500 192.168.1.1 interface Fa5/1
```

Note: When you enable both IP Source Guard and port security by using the **ip verify source port-security** command, there are two caveats - the DHCP server must support option 82, or the client is not assigned an IP address and the MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.

In order to verify this powerful security feature, use:

show ip verify source

This command shows the enabled ports and current addresses.

Common Issues with IP Source Guard

Common problems to watch out for include:

- DHCP Snooping is not active or the database is stale.
- IP Source Guard is not enabled on the correct ports.

- Static mappings are not properly constructed.

Private VLANs

Private VLANs (PVLANs) allow you to provide coarse access control within a VLAN to limit connectivity. Thanks to this feature, a VLAN can be divided into multiple logical segments (secondary VLANs) that have specific connectivity requirements. The secondary VLANs can create host groups or isolate individual hosts and still provide Layer 3 routing out of the VLAN.

The isolation capabilities that Private VLANs provide eliminates the need for a separate VLAN and IP subnet per host group and provides a coarse access control mechanism that is easy to manage.

PVLANs introduce the concept of a primary VLAN, which is the VLAN to which all devices belong. This VLAN hosts a particular OSI Layer 3 subnet. Inside this VLAN, you can create the secondary VLANs.

Private VLANs function as a result of PVLAN port types. They are as follows:

- Isolated ports – these ports can only communicate with the promiscuous port
- Promiscuous port – this port has the ability to communicate with all other ports
- Community ports – these ports can communicate with other members of the community, and with the promiscuous port

Notice that an isolated port has complete Layer 2 separation from other ports within the same primary PVLAN, except from promiscuous ports. The promiscuous port is where you typically connect routers or other Layer 3 devices to permit the isolated device to communicate off of the primary VLAN. PVLANs block all traffic to isolated ports, except the traffic from promiscuous ports.

Follow these steps to configure Private VLANs (PVLANs):

Step 1. Set the VTP mode on all involved switches to transparent.

```
vtp mode transparent
```

2. Create the (isolated and community) secondary VLANs.

```
vlan 600
 private-vlan community
vlan 400
 private-vlan isolated
```

3. Create the primary VLAN.

```
vlan 200
 private-vlan primary
```

4. Associate secondary VLANs to the primary VLAN. You can map only one isolated VLAN to a primary VLAN, but you can map more than one community VLAN to a primary VLAN.

```
vlan 200
 private-vlan association 400,600
```

5. Configure an interface to be an isolated or community port by associating it with the appropriate secondary and primary VLAN.

```
interface FastEthernet 5/1
 switchport mode private-vlan host
 switchport private-vlan host-association 200 400
```

6. (Optional) Configure an interface as a promiscuous port, associating it with all the secondary VLANs and the primary VLAN.

```
interface FastEthernet 5/4
 switchport mode private-vlan promiscuous
 switchport private-vlan mapping 200 400,600
```

7. (Optional) In multiswitch PVLAN environments, configure PVLAN functionality on the interswitch trunk ports.

```
interface GigabitEthernet0/1
 switchport mode trunk
 switchport trunk encapsulation dot1q
```

As you might guess, there are many commands available in order to verify the Private VLAN infrastructure. In order to verify the private VLAN types in place on the device, use the command:

```
Switch#show vlan private-vlan type

Vlan Type

---- ---------

200 primary

400 isolated

600 community
```

In order to verify the PVLAN associations, use:

```
Switch#

Switch#show vlan private-vlan

Primary Secondary Type Ports

------- --------- ---------------- ------------------------------

200 400 isolated fa5/1, fa5/4

200 600 community fa5/2, fa5/3, fa5/4
```

Common Issues with PVLANs

Keep in mind the following potential issues when configuring your PVLANs:

- An isolated or community VLAN can have only one primary VLAN associated with it.
- VLAN Trunking Protocol (VTP) does not support PVLANs. Manually configure PVLANs on each device on which you want PVLAN ports.
- For PVLANs, the VTP mode must be transparent, not client or server.
- Be sure to verify each step of the complex configuration.
- Be sure to verify the correct use of secondary VLANs (isolated versus community).

Chapter Challenge: Catalyst Security Trouble Tickets

The following section includes sample trouble tickets for Catalyst security. Remember to load the appropriate initial configurations for these sample tickets. Here is the simple topology that will be used for these tickets:

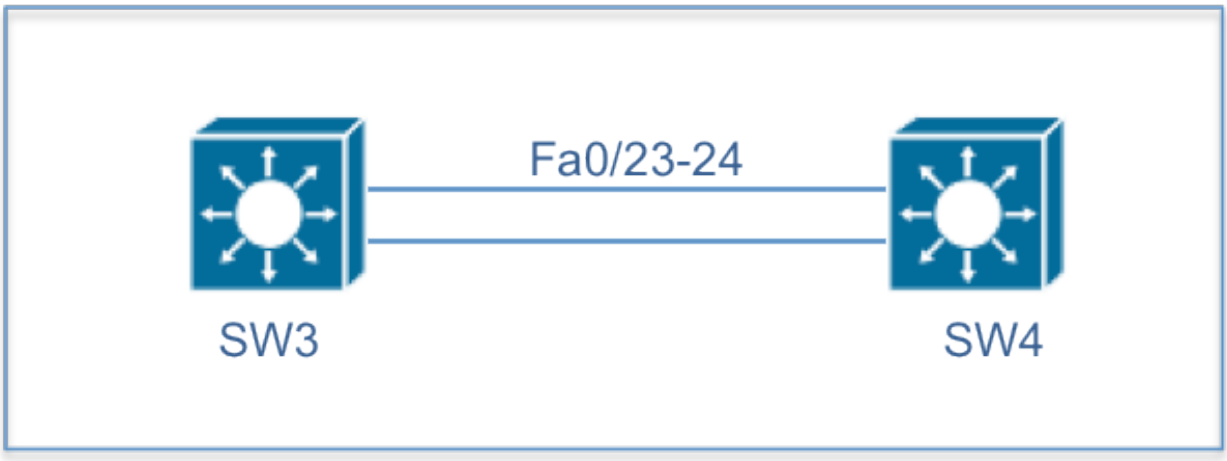


Figure 5-1 A Sample Troubleshooting Topology

Trouble Ticket #1

Your IT manager is concerned with a new attack that has been reported. Ensure your Junior Administrator configured SW4 so that its Fa0/10 port will shut down should incoming VLAN-tagged packets smaller than 67 bytes arrive at a rate of 5000 packets per second or higher.

2 points

Chapter Challenge: Catalyst Security Trouble Ticket Solutions

The following section includes solutions to the Trouble Tickets associated with Catalyst security.

Trouble Ticket #1

Your IT manager is concerned with a new attack that has been reported. Ensure your Junior Administrator configured SW4 so that its Fa0/10 port will shut down should incoming VLAN-tagged packets smaller than 67 bytes arrive at a rate of 5000 packets per second or higher.

2 points

Step 1 - Fault Verification:

What is the current configuration for protection against small frames?

```
SW4#show run | include small
 small-frame violation-rate 5
SW4#
```

The current configuration has two issues! The value for the packet per second rate is misconfigured, and we need to enable the error disabling of a port based on this condition. Note int his case it was a default, but someone reversed that.

Let us check the correct value syntax:

```
SW4(config)#int fa0/10
SW4(config-if)#small-frame violation-rate ?
  <1-10000>  Packets per second
```

Step 2 – Fault Remediation

Here is the correct configuration for this scenario:

```
SW4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW4(config)#errdisable detect cause small-frame
SW4(config)#interface fa0/10
SW4(config-if)#small-frame violation 5000
SW4(config-if)#no shutdown
SW4(config-if)#end
SW4#
```

This is a perfect example of when intimate knowledge of where things are in the DOC-CD might be required in a Lab Exam. This feature syntax might be difficult to remember, but very easy to find in the DOC-CD.

Step 3 – Fault Remediation Verification

Here, verification is simple:

```
SW4#show errdisable detect
ErrDisable Reason      Detection    Mode
-----------------      ---------    ----
arp-inspection          Enabled      port
bpduguard               Enabled      port
channel-misconfig       Enabled      port
community-limit         Enabled      port
dhcp-rate-limit         Enabled      port
dtp-flap                Disabled
gbic-invalid            Enabled      port
inline-power            Enabled      port
invalid-policy          Enabled      port
l2ptguard               Enabled      port
link-flap               Enabled      port
loopback                Enabled      port
lsgroup                 Enabled      port
mac-limit               Enabled      port
pagp-flap               Enabled      port
port-mode-failure       Enabled      port
psecure-violation       Enabled      port/vlan
security-violation      Enabled      port
sfp-config-mismatch     Enabled      port
small-frame             Enabled      port
storm-control           Enabled      port
udld                    Enabled      port
vmps                    Enabled      port
SW4#show run int fa0/10
Building configuration...

Current configuration : 67 bytes
!
interface FastEthernet0/10
 small-frame violation-rate 5000
end

SW4#
```

Additional Trouble Tickets and Chapters are coming soon from our Editors. Any questions please contact Anthony Sequeira at compsolv@me.com