INTERNETWORK

e pert

CCIE Routing & Switching Advanced Troubleshooting Bootcamp

Troubleshooting Overview

http://www.INE.com

Instructor Introduction

- Brian McGahan, CCIE #8593
  - MCSE NT 4.0, CCNA, CCNP
  - CCIE Routing and Switching - 2002
  - CCIE Service Provider – 2006
  - CCIE Security – 2007
  - bmcgahan@ine.com
- Anthony Sequeira, CCIE #15626
  - CCIE Routing and Switching
  - asequeira@ine.com

Copyright © 2009 Internetwork Expert, Inc
www.INE.com

e pert

Online Classroom Introduction

- Classroom software overview
  - Bandwidth settings
- How questions are handled during class
  - Questions of interest to the whole class
  - Questions of interest to only you
  - NDA related questions
- What to do if you have a technical issue during class
  - US and Canada: 1-877-224-8987 ext 2
  - International: +1-775-825-9943

Questions

- Cisco NDA Agreement
- Questions In Class
  - Participation is key
- Offline Questions
  - Blog
    - http://blog.INE.com
  - Online Community
    - http://www.IEOC.com
    - Web forum / mailing lists

Class Timing

- Start daily at 7am PDT (GMT -8)
- 10 minute break ~ every 50 minutes
- 1 hour lunch break at 10am PDT
- Class ends ~ 4pm PDT

Class Schedule

- Day 1
  - Introduction
  - CCIEv4 Changes Overview
  - Troubleshooting Overview
  - Layer 2 Troubleshooting
- Day 2
  - Layer 3 Troubleshooting
- Day 3
  - Layer 3 Troubleshooting (cont.)
  - Layer 4 – 7 Troubleshooting
    - Security, Management, IOS Features, etc.
- Day 4
  - Full Scale Troubleshooting Lab
- Day 5
  - Full Scale Lab Breakdown
  - Class Review

CCIE R&S Version 4 Changes

- As of October 18, 2009, CCIE R&S Exam format undergoes a major format change
- Three lab exam sections:
  - Short Answer / OEQ's – 30 minutes
  - Troubleshooting – 2 hours
    - Our focus for this class
  - Configuration – 5 hours 30 minutes
- Candidates must pass all three sections to pass the lab

Short Answer / OEQ's Section

- Four "Open Ended" questions
- 30 minutes allotted
- Candidates must answer 3 out of 4 correctly to pass
- Answers typically one sentence or less

Troubleshooting Section

- ~ 8 – 12 "Trouble Tickets"
- ~ 20 – 25 points total
- 2 hours allotted
  - Extra time can be applied to Configuration section
- Assume 80% to pass
- DocCD access allowed

Troubleshooting Section (cont.)

- Each ticket is independent of others, and can be solved in any order
  - Implies large topology
- Only working configurations are correct
  - i.e. results oriented like much of Configuration Section
- No Layer 1 troubleshooting
  - e.g. Fiber cabling issues

Troubleshooting Topology

- Uses IOS on Unix (IOU) for virtual hardware topology
  - Router hardware emulator like Dynamips, not an IOS "simulator"
  - Nothing special from our perspective, just an IOS instance
- IOU does not support Catalyst IOS
  - Implies no Layer 2 switching troubleshooting

Configuration Section

- ~ 70 – 76 points total
- Assume 80% to pass
- Still main focus of the exam
- Less "basic" configuration and more pre-configuration
  - e.g. access VLAN assignments
- Pre-configuration can include Layer 2 switching troubleshooting

New CCIEv4 Topics

- MPLS
  - Basic L3VPN, no L2VPN / MPLS TE / QoS / etc.
- OER / PfR
- IPv6 Multicast
- IPv6 EIGRP
- Zone Based Policy Firewall
- IOS IPS
- Not all topics on all exams or in all *sections* of exams
  - Doesn't mean you can shortcut them though

What Is Troubleshooting?

- Per Wikipedia… "a form of problem solving most often applied to repair of failed products or processes. It is a logical, systematic search for the source of a problem so that it can be solved, and so the product or process can be made operational again."
- The key is that troubleshooting is *logical* and *systematic*
- Fixing a problem by dumb luck does not constitute troubleshooting

Why Troubleshooting?

- Today's networks are more high-availability minded than ever, and downtime means loss of revenue in…
  - Employee productivity
  - Customer SLA violations
  - Regulatory fines
  - Etc.
- One key way expert-level engineers set themselves apart from average engineers is troubleshooting methodology
  - average engineer runs around like a chicken with its head cut off
  - expert engineer keeps a cool head and follows a structured approach
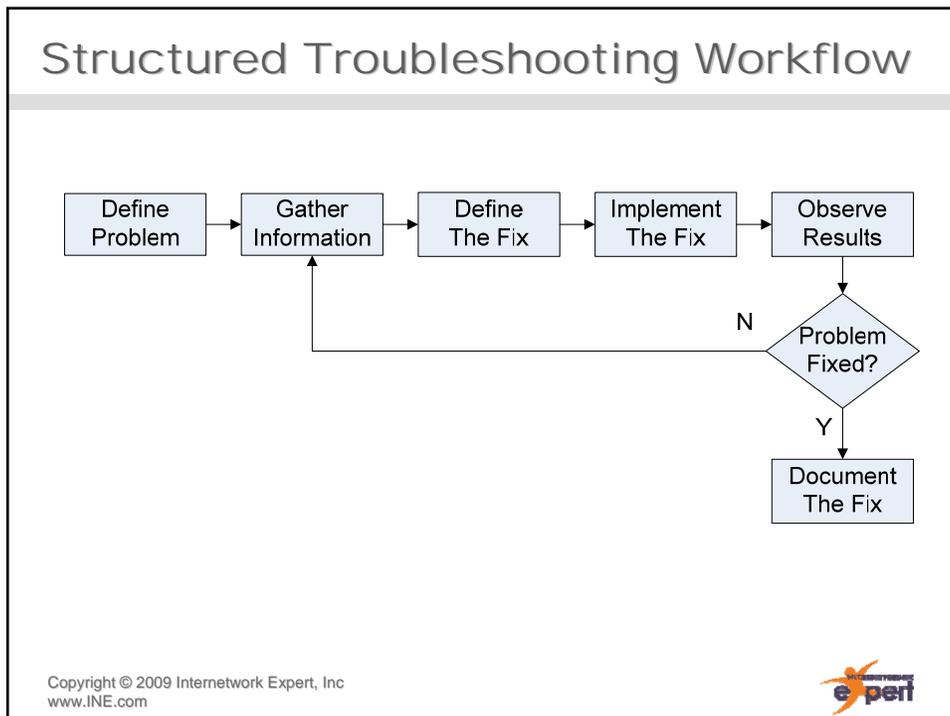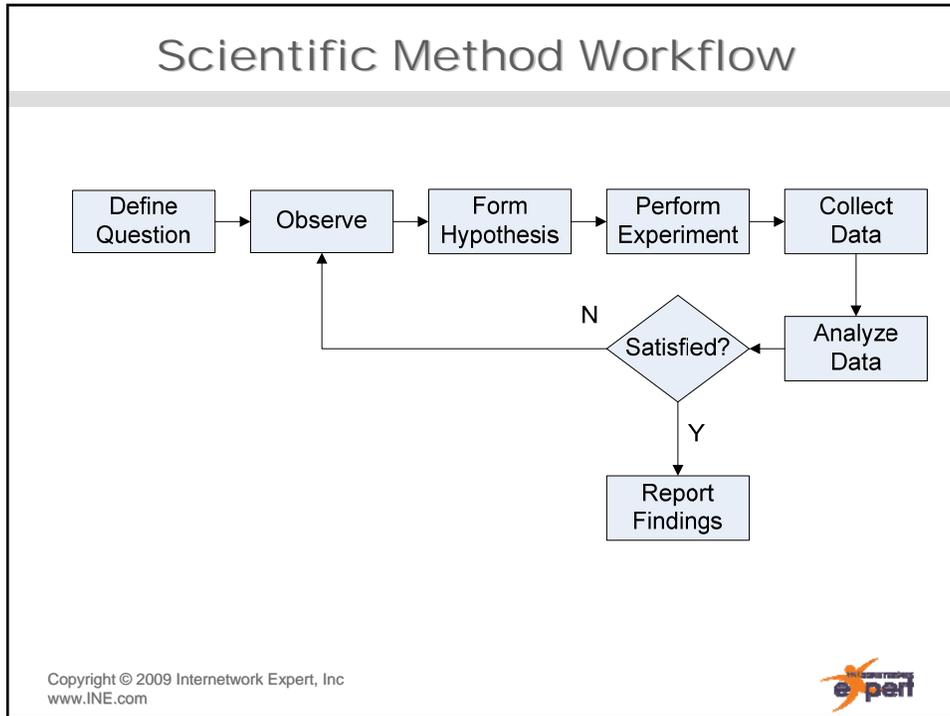
Structured Troubleshooting Approach

- Defines a logical and systematic method of troubleshooting that can be applied to any case
  - E.g. troubleshooting VoIP call quality and OSPF neighbor adjacency involves different discrete steps, but logical approach is the same
- Structured troubleshooting is closely analogous to the Scientific Method of conducting experiments

Scientific Method Workflow

```
[Define Question] → [Observe] → [Form Hypothesis] → [Perform Experiment] → [Collect Data]
                       ↑                                                          ↓
                       |                          N                        [Analyze Data]
                       ←───────────────────── <Satisfied?> ←──────────────────┘
                                                    |
                                                    Y
                                                    ↓
                                             [Report Findings]
```

Structured Troubleshooting Workflow

```
[Define Problem] → [Gather Information] → [Define The Fix] → [Implement The Fix] → [Observe Results]
                        ↑                                                                  ↓
                        |                               N                          <Problem Fixed?>
                        ←──────────────────────────────────────────────────────────────┘
                                                                                         |
                                                                                         Y
                                                                                         ↓
                                                                                  [Document The Fix]
```

Defining The Problem

- Network problems are generally discovered in two ways
  - Reactive
    - e.g. users submit tickets to the help desk that web browsing is slow
  - Proactive
    - e.g. SNMP reports a linkdown event
- In either case, more investigation is needed to find the root of the cause

Gathering Information

- Apart from asking users for more information on tickets submitted, gathering information is in the form of…
  - show commands
  - debug commands
    - Typically not used in real-world unless network-down emergency
  - Misc. testing tools
    - PING
    - Traceroute
    - Telnet
    - Etc.
- Ultimate goal is to isolate the issue as closely as possible

How To Gather Information

- Structured troubleshooting involves isolating the operation network into functional layers
  - E.g. OSI Model or TCP/IP Model
- Where to actually start isolating is a personal preference
  - Common approaches are…
    - Top-Down
    - Bottom-Up
    - Divide and Conquer
- Key to remember is that layers have a cascading effect
  - E.g. if physical layer (i.e. layer 1) is down, all layers above it are broken

Top Down Troubleshooting

- Most useful for application related issues
  - E.g. user can't send email – start by checking their email settings
- Within the scope of CCIE lab troubleshooting, would be very time consuming

Bottom Up Troubleshooting

- Verify each layer starting with physical and proceed to the next
  - Is the link UP/UP?
  - Are the layer 2 options correct?
  - IP properly configured?
  - IGP adjacency exists?
  - Etc.
- Like top-down, can be very time consuming depending on where the problem actually lies

Divide and Conquer

- Goal is to reduce search time by picking a layer to start at
- Based on results of testing, further verification goes either up or down the stack
- E.g. for troubleshooting email problem…
  - Can I ping the mail server?
    - If yes, go up the stack
    - If no, go down the stack

Defining & Implementing The Fix

- Ideally up to this point the issue is sufficiently isolated to make an educated guess as to how the problem can be fixed
- Proper "Change Control" at this stage is key
  - Clearly define the proposed fix
  - Implement the proposed fix
  - Did it work?
    - If yes, proceed forwards
    - If no, roll back
- Changing too many variables at once can compound the problem even further

Observing The Results

- Depending on the nature of the problem, verification of the solution can be either straightforward or complicated
  - E.g. user said they couldn't email, now they can, problem straightforward and solved
  - E.g. users experienced low VoIP quality, quality is now good, but only time will tell
- Within the scope of CCIE lab exam we can assume that verification will be concrete

Reiteration

- If the problem was not solved, a further dilemma occurs
  - Did I misdiagnose the problem in the first place?
  - Are there significant variables that were overlooked?
  - Was my fix not appropriate?
- Before making further changes, more information should be gathered
  - Did the situation change since I implemented a fix?
    - If yes, for the better or worse?
    - If not, why not?