Enterprise Linux Security Administration

Course Summary

Description

This highly technical course focuses on properly securing machines running the Linux operating systems. A broad range of general security techniques such as user/group policies, and file integrity checking are covered. Advanced security technologies are taught such as Kerberos, SELinux, and the hardening of popular applications such as Apache, databases, and email systems. At the end of the course, students have an excellent understanding of the potential security vulnerabilities -- know how to audit existing machines, and best practices how to securely deploy new Linux servers.

Topics

- Security Concepts
- Probing, Mapping and Scanning for Vulnerabilities
- Password Security and PAM
- Secure network time protocol (NTP)
- Kerberos Concepts
- Kerberos Components
- Implementing Kerberos
- Administrating and Using Kerberos
- Securing the filesystem
- Tripwire
- Securing Apache
- Securing PostgreSQL
- Securing EMail Systems
- SELinux Concepts
- SELinux Policy

Audience

This course is designed for Linux system administrators.

Prerequisites

Individuals planning to take this class should have strong Linux system administration experience. Students should be comfortable with concepts and tasks such as editing text files in UNIX and starting and stopping services/daemons. A good grasp of networking concepts will be helpful.

Duration

Five days

ProTech
protechtraining.com

"Charting the Course ...

... to Your Success!"

Enterprise Linux Security Administration

Course Outline

I. Security Concepts
- A. Basic Security Principles
- B. RHEL/FC/SLES/SL Default Install
- C. RH/SUSE Firewall Options and File Security
- D. Minimization – Discovery
- E. Service Discovery
- F. Hardening
- G. Security Concepts
- H. Lab 1 - Security Concepts
- I. Discovering what software packages are installed and removing unneeded packages
- J. Using lokkit for firewall configuration
- K. Identification of running services and removing unneeded services
- L. Increasing security using system calls and chroot

II. Probing, Mapping and Scanning for Vulnerabilities
- A. The Security Environment
- B. Stealth Reconnaissance
- C. The WHOIS database
- D. Interrogating DNS
- E. Discovering Available Hosts and Applications
- F. Reconnaissance with SNMP
- G. Discovery of RPC Services
- H. Enumerating NFS Shares
- I. Nessus Insecurity Scanner and Installation
- J. Lab 2 - Probing, Mapping and Nessus
- K. Discovery of listening services and remote stack fingerprinting
- L. Installing, configuring and testing Nessus insecurity scanner

III. Password Security and PAM
- A. UNIX Passwords
- B. Password Aging
- C. Auditing Passwords
- D. PAM Implementation, Management, and Control Statements
- E. PAM Modules
- F. pam_stack.so, pam_unix.so, pam_unix2.so, pam_cracklib.so, pam_pwcheck.so, pam_env.so, pam_xauth..so, pam_tally.so, pam_wheel.so, pam_limits.so, pam_nologin.so, pam_deny.so, pam_securetty.so, pam_time.so, pam_access.so, pam_listfile.so, pam_lastlog.so, pam_warn.so, pam_console.so, pam_resmgr.so, and pam_devperm.so
- G. User Device Access: resmgr
- H. Lab 3 - Pluggable Authentication Modules
- I. Auditing user password quality
- J. Creating additional dictionaries for use with cracklib
- K. Working with PAM modules
- L. Limiting access activities of users and accounts

IV. Secure network time protocol (NTP)
- A. The Importance of Time
- B. Time Measurements and Synchronization Methods
- C. NTP Evolution
- D. Time Server Hierarchy
- E. perational Modes
- F. NTP Clients
- G. Configuring NTP Clients and Servers
- H. Securing NTP
- I. NTP Packet Integrity
- J. Useful NTP Commands
- K. Lab 4 - Secure NTP
- L. Configuring NTP peering
- M. Configuring strong authentication on a NTP server
- N. Defining Access Control Lists (ACLs) for secure access to NTP server

V. Kerberos Concepts
- A. The Computing Landscape
- B. Common Security Problems
- C. Account Proliferation
- D. The Kerberos Solution
- E. Kerberos History, Implementations, and Concepts
- F. Kerberos Principals, Safeguards, and Components

Enterprise Linux Security Administration

Course Outline (cont'd)

G. Authentication Process and Identification Types
H. Logging In
I. Gaining and Using Privileges

VI. Kerberos Components
A. Kerberos Components
B. Kerberos Principal Review
C. Kerberized Services Review and Clients
D. KDC Server Daemons
E. Configuration Files
F. Utilities Overview
G. Kerberos SysV Init Scripts

VII. Implementing Kerberos
A. Plan Topology and Implementation
B. Kerberos 5 Client and Server Software
C. Synchronize Clocks
D. Creating and Configuring the Master KDC
E. KDC Logging
F. Specifying [realms] and [domain_realm]
G. Allow Administrative Access
H. Create KDC Databases and Administrators
I. Install Keys for Services and Start Services
J. Add Host Principals and Common Service Principals
K. Configure Slave KDCs
L. Client Configuration
M. Install krb5.conf on Clients
N. Client PAM Configuration
O. Install Client Host Keys
P. Lab 7-Implementing Kerberos
Q. Configuring a master KDC
R. Configuring a slave KDC
S. Configuring a Kerberos client

VIII. Administrating and Using Kerberos
A. Administrative Tasks
B. Key Tables
C. Managing Keytabs
D. Principals and Managing Principals
E. MIT Principal Policy
F. Viewing Principals
G. MIT Managing Policies
H. Goals for Users
I. Signing Into Kerberos
J. Ticket types and Viewing Tickets
K. GUI Kerberos Ticket Management
L. Removing Tickets
M. Passwords and Changing Passwords
N. Giving Others Access
O. Using Kerberized Services
P. Kerberized FTP
Q. Enabling Kerberized Services
R. OpenSSH and Kerberos
S. Lab 8 - Using Kerberized Clients
T. System configuration for use of kerberized client and server applications
U. Using the kerberized telnet to connect via a ticket and encrypt the data for the session
V. Exploring the utility and behavior of forwardable tickets
W. Configuring an OpenSSH server and client to accept and use Kerberos Authentication
X. Testing Kerberos authentication with OpenSSH

IX. Securing the filesystem
A. Filesystem Mount Options
B. NFS Properties and NFS Export Option
C. NFSv4 and GSSAPI Auth
D. Implementing NFSv4
E. File Encryption with GPG and OpenSSL
F. Encrypted Loopback FS
G. Lab 9 - Filesystem Security, and File Encryption
H. Modification of filesystem mounting options to increase system security
I. Configuring and securing an NFS share
J. Encrypting and decrypting files using GPG and openssl
K. Setting up a NFSv4 share with GSSAPI/Kerberos authentication

X. Tripwire
A. Host Intrusion Detection
B. Using RPM as an IDS
C. TripWire History and Concepts
D.

Enterprise Linux Security Administration

Course Outline (cont'd)

E. TripWire Installation, Policies, and Configuration
F. TripWire Commands and General Operation
G. Lab 10 - File integrity checking with rpm / TripWire
H. Verifying the integrity of files on the system and files in a directory
I. Configuring TripWire to monitor files and report changes

XI. Securing Apache
A. Apache Overview
B. RH/SUSE Default Configuration
C. Configuring CGI
D. Turning off unneeded modules
E. Configuration Delegation and Scope
F. ACL by IP Address
G. HTTP User Authentication
H. Standard Auth Modules
I. HTTP Digest Authentication
J. Authentication via SQL, LDAP, and Kerberos
K. Scrubbing HTTP Headers
L. Metering HTTP Bandwidth
M. Lab 11- Securing Apache
N. Increasing security and optimizing Apache by disabling unneeded modules
O. Removing Apache and PHP version from HTTP headers
P. Setting up virtual hosts
Q. Creating CGI scripts to "deface" another's files and setting permissions against exploit
R. Showing files can be read by virtual host users and employing "suexec" to protect against access
S. Configuring and testing mod_auth_kerb

XII. Securing PostgreSQL
A. PostgreSQL Overview and Default Configuration
B. Configuring SSL
C. Authentication Methods and Advanced Authentication
D. Ident-based Authentication
E. Lab 12- Securing PostgreSQL
F. Configuring PostgreSQL to accept remote TCP connections
G. Configuring PostgreSQL to support strong authentication via SSL
H. Configuring PostgreSQL to support Kerberos
I. Setting up and configuring a web based multi-user PHP calendaring application that uses PostgreSQL
J. Configuring Apache to support Kerberos authentication and to require SSL

XIII. Securing EMail Systems
A. SMTP Overview and Implementations
B. Selecting an MTA
C. Security Considerations
D. Postfix Overview
E. Chrooting Postfix
F. Connections and Relays
G. SMTP AUTH & StartTLS/SSL
H. Secure Cyrus IMAP Config
I. Using GSSAPI/Kerberos Auth
J. Lab 13 - Securing Email
K. Configuring a system to use Postfix
L. Configuring Postfix to listen on the network and accept mail
M. Modifying Postfix's SysV Init script to setup and maintain the proper environment for chrooting Postfix daemons each time it starts
N. Configuring Postfix to chroot some of its daemons
O. Configuring Postfix to use SMTP AUTH via PAM for secure relaying
P. Configuring Postfix to support STARTTLS to secure SMTP AUTH
Q. Configuring Cyrus IMAP with SSL/TLS for IMAPS and POP3 access
R. Configuring Postfix to deliver mail to Cyrus IMAP
S. Setting up Evolution to test Postfix and Cyrus IMAP
T. Generating Kerberos principals for Cyrus IMAP and Postfix
U. Re-Configuring Cyrus IMAP and Postfix to perform GSSAPI/Kerberos authentication

Enterprise Linux Security Administration

Course Outline (cont'd)

V. Re-Configuring Evolution to preform GSSAPI/Kerberos authentication

XIV. SELinux Concepts

A. DAC vs. MAC
B. Shortcomings of Traditional UNIX Security
C. SELinux Goals, Terms, and Logical Architecture
D. SELinux in Action
E. Activating and Interfacing SELinux
F. SELinux Commands and Roles
G. Modified System Utilities
H. Lab 14 - SELinux Concepts
I. Installing and initializing SELinux
J. Working with several SELinux management commands to see how roles and contexts are used on the system

XV. SELinux Policy

A. SELinux Policies Review
B. Choosing a Policy
C. Compiled Policy Files
D. Policy Source Files
E. M4 Macro Language
F. File Context Files (*.fc)
G. Type Enforcement Files (*.te)
H. Booleans
I. Graphical Policy Tools
J. Policy Analysis
K. Policy Customization
L. Troubleshooting SELinux Problems
M. Lab 15 - SELinux Policy
N. Enabling Strict Policy
O. Changing roles on the system
P. Understanding the difference between how context labels are treated with the cp and mv commands
Q. Setting SELinux Boolean Values
R. Modifying the default policy so that users can do a directory listing in /var/log