12.6.4 Linux Firewall Facts

A firewall basically establishes a barrier between the internal network, which is assumed to be secure and trusted, and the external network, which is usually the internet and is not secure or trusted. Most operating systems, including Linux, offer software-based firewalls to protect networks and systems from external threats.

This lesson covers the following topics:

- Access control lists
- Popular Linux firewalls
- Other Linux firewall considerations

Access Control Lists

ACLs are the rules a firewall uses to process IP packets. Linux firewall technologies often use different methods to organize their configuration options, but ACL concepts are still at the heart of a firewall's design.

- ACLs determine whether routed packets are accepted, rejected, or dropped.
  - Accepted packets are forwarded on to their destinations.
  - Rejected packets are blocked, and a message is sent back to the packet's sender.
  - Dropped packets are also blocked, but no message is sent.
- ACLs are stateless firewall filters.

  A stateful firewall looks at traffic patterns, tunneling, and encryption to determine how to filter packets.

- ACLs use the following packet characteristics to determine how to filter a packet.
  - Source address
  - Destination address
  - Ports
  - Protocols
- ACLs can log each time they're used to filter a packet.
- The packets that are the target of ACLs can be captured by setting a logging option.

Popular Linux Firewalls

There are many third-party Linux firewalls, and a few of them are included in major Linux distributions. Most of them are based on Netfilter.

- Netfilter is part of the Linux kernel.
- Netfilter is used for network address translation and port translation.
- Netfilter supplies the kernel's IP packet filtering functions, which are used by firewall applications.
- Linux firewall applications interface with Netfilter to perform firewall functions.

The following are popular Linux firewalls that are based on Netfilter.

Firewall	Description

IPTables	IPTables is a firewall application that's pre-installed on most Linux distributions.
	- IPTables is a rule-based front-end tool that interfaces with Netfilter to decide which packets to filter.
	- Internally, IPTables consists of five pre-defined tables that contain chains.
	  - The kernel accesses each chain at a specific point while processing an IP packet.
	  - Each chain has its own purpose and contains rules.
	- Configure an IPTables firewall by adding, deleting, and customizing the rules contained in each chain.
	  - For a basic firewall, only the INPUT, FORWARD, and OUTPUT chains in the filter table are modified.
	
	IP packets are filtered according to the IPTables rules.
	- Each chain's rules are traversed in order.
	- Each rule has fields that are matched against the IP packet.
	  - If a match is made, the action in the rule is taken. No other rules in the chain are checked.
	  - If the packet doesn't match the rule, the rule is skipped, and the next rule is checked.
	  - Normally, the last rule is configured with wildcards so that it matches any packet. In many cases, the action for the last rule is to reject the packet.

Uncomplicated Firewall	Uncomplicated Firewall (UFW) provides a user-friendly framework for managing Netfilter. - A command line interface is provided to configure the firewall.

	▪ A few GUI tools make working with the UFW incredibly simple.

firewalld	The firewalld firewall is pre-installed on many Linux distributions.
	▪ firewalld is a front-end controller for IPTables.
	▪ With firewalld, the IPTables commands are still available.
	▪ firewalld has both a command line and graphical interface.
	▪ firewalld uses zones and services instead of chains and rules.
	  ▪ Zones are pre-constructed rulesets for various trust levels.
	  ▪ Different zones allow different network services, ports, protocols, and incoming traffic types, while denying everything else.
	  ▪ You can apply a zone to different network interfaces.
	  ▪ You can configure firewalld with rules to allow traffic for specific network services.
	  ▪ You can add custom service rules to any zone.

Other Linux Firewall Considerations

The following are items to consider when implementing a Linux firewall.

Firewall Consideration	Description

IP Forwarding	IP forwarding is another name for routing. It's sometimes called kernel IP forwarding because it's a feature of the Linux kernel.
	▪ Enable IP forwarding by writing a 1 to the ip_forward file.
	  ▪ Enable IPv4 forwarding by writing to the /proc/sys/net/ipv4/ip_forward file.
	  ▪ Enable IPv6 forwarding by writing to the /proc/sys/net/ipv6/ip_forward file.
	▪ Be cautious about enabling IP forwarding without a firewall, especially if an interface connects to the internet or to a subnet you don't control.

Dynamic Rule Sets	Dynamic rule sets automate the rules IPTables use to filter network traffic and prevent intrusions.
	▪ There are two popular Python scripts that are classified as intrusion prevention software.
	  ▪ DenyHosts
	  ▪ Fail2ban
	▪ Both scripts monitor log files and react to common security problems, such as brute-force attacks, by adding or modifying firewall rules.

IPset	IPset is a companion application to IPTables that allows you to easily set firewall rules for a block of IP addresses.
/etc/services	Many firewall applications read from the /etc/services file. This file is a list of well-known services and their port assignments. When you update firewall rules, consider updating this file with new services and ports.
privileged ports	The ports from 1 to 1023 are privileged ports. ▪ Only the root account has access to ports 1 to 1023. ▪ Privileged ports give confidence in internal networks where only trusted individuals have passwords to the root account. ▪ Internal firewalls may be more tolerant when passing network traffic using these ports.