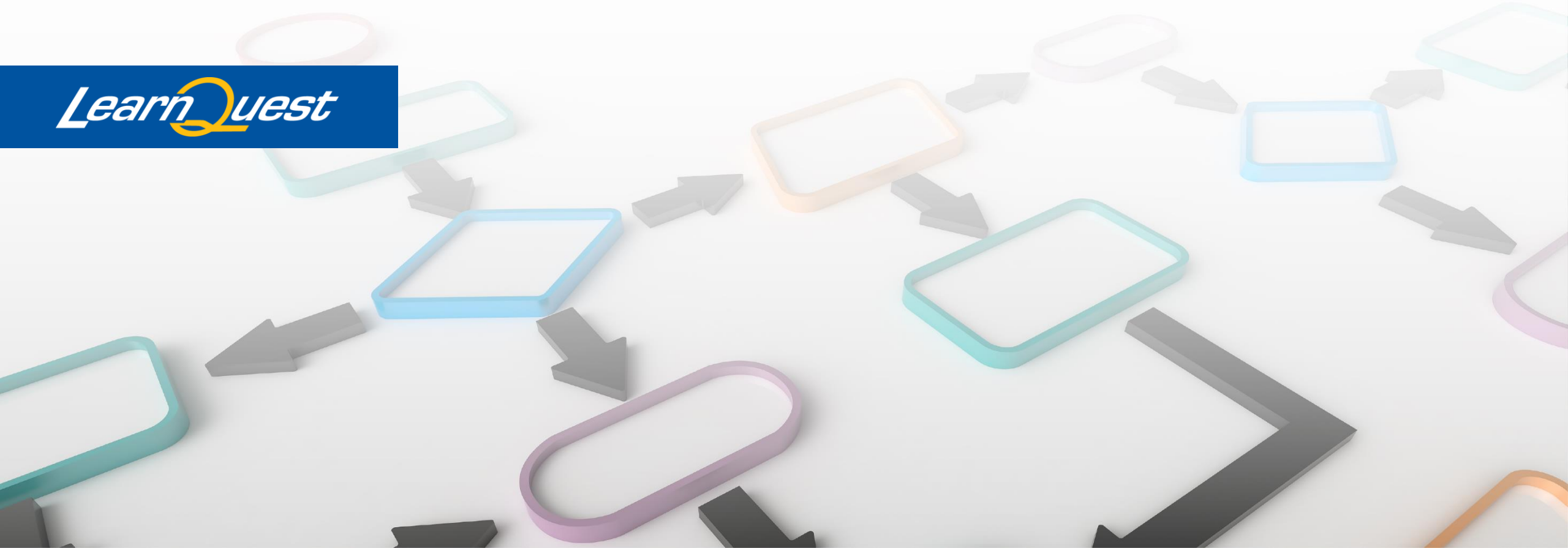
Securing Linux Systems

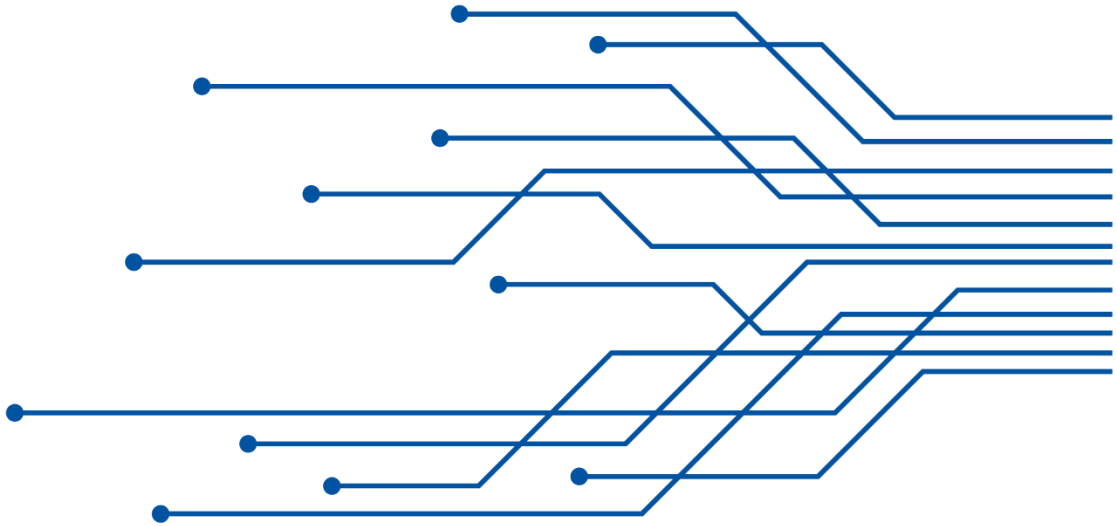
- 3rd Course in Linux Foundations Specialization



Administer Access and Authentication

In this module, we look at ways to increase our Linux Security.

2



Learning Objectives

Administer Access and Authentication

Upon completion of this module, learners will be able to:

- Utilize PAM to enforce strong passwords
- Describe Public Key Infrastructure (PKI)
- Use Secure Shell (SSH)
- Set up a Virtual Private Network (VPN)

Lesson 1

PluggableAuthentication Modules (PAM)

In this lesson, we look at how to utilize PAM to enforce strong passwords

Pluggable Authentication Modules

- Pluggable Authentication Modules (PAMs) provide centralized authentication services for Linux and applications.
- PAM configuration files are located in the /etc/pam.d/ directory.
- The records in a PAM configuration file have the following syntax:
  - TYPE CONTROL-FLAG PAM-MODULE [MODULE-OPTIONS]

TYPES in PAM

account - Implements account validation services. For example: enforcing time of day restrictions as well as determining if the account has expired.

auth - Provides account authentication management services. For example: prompting for a password and verifying the password.

password - Manages account passwords. For example: minimum password lengths and limiting incorrect password entry attempts.

session - Provides authenticated account session management for session start and session end. For example: logging when the session began and ended as well as mounting the account's home directory.

CONTROL-FLAG in PAM

include - Adds status codes and response ratings from the designated PAM configuration files into the final status.

optional - Conditionally adds the module's status code to the final status. If this is the only record for the PAM service type, it is included. If not, the status code is ignored.

requisite - If the module returns a fail status code, a final fail status is immediately returned to the application without running the rest of the modules within the configuration file.

required - If the module returns a fail status code, a final fail status will be returned to the application, but only after the rest of the modules within the configuration file run.

substack - Forces the included configuration files of a particular type to act together returning a single status code to the main module stack.

sufficient - If the module returns a success status code and no preceding stack modules have returned a fail status code, a final success status is immediately returned to the application without running the rest of the modules within the configuration file. If the module returns a fail status code, it is ignored.

PAM Modules

pam_unix.so module - performs authentication using account and password data stored in the /etc/passwd and /etc/shadow files.

pam_pwhistory.so module - checks a user's newly entered password against a history database to prevent a user from reusing an old password.

pam_pwquality.so - can enforce rules for new passwords. For example: setting a minimum password length.

pam_tally2.so and pam_faillock.so modules - implement account lockout.

pam_securetty.so module - restrict root account logins

Lesson 1 Review

PAM provides centralized authentication services for Linux and applications

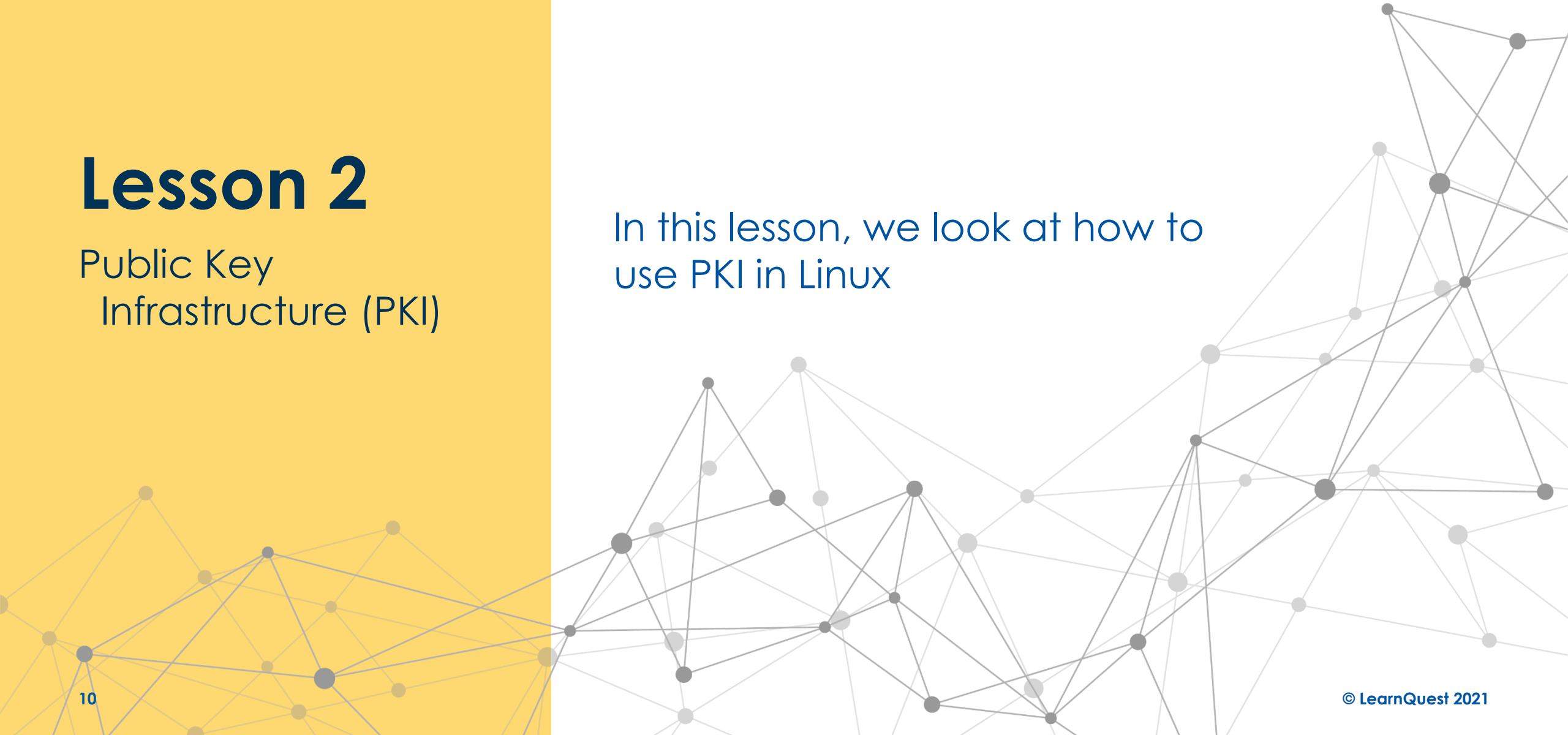
PAM Modules can enforce password rules

PAM Modules can lockout accounts

Lesson 2

Public Key Infrastructure (PKI)

In this lesson, we look at how to use PKI in Linux



Public Key Infrastructure (PKI)

Public Key Infrastructure (PKI) is a use of cryptography for authenticating users and devices.

A trusted party digitally signs a document certifying that a particular cryptographic key belongs to a particular user or device.

The key can then be used as an identity for the user in digital networks.

Certificates

A Certificate authority (CA) verifies a person's identity and issues a digital certificate to the requesting person.

The digital certificate provides identification proof with an embedded key.

The certificate's key is used to encrypt data and sign it.

You can generate and sign your own certificate for testing.

This type of certificate is called a self-signed digital certificate.

Cryptographic Keys

Private Keys or Symmetric keys encrypt data using a cryptographic algorithm and a single key. Plain text is both encrypted and decrypted using the same key.

Symmetric key cryptography is very fast. Unfortunately, you must share the private key to allow someone to decrypt the data.

Public/Private Key Pairs or Asymmetric keys encrypt data using a cryptographic algorithm and two keys. The public key is used to encrypt the data and the private key decrypts the data.

Lesson 2 Review

Symmetric key cryptography is very fast

Asymmetric keys encrypt data using a cryptographic algorithm and two keys.

PKI uses the public key to encrypt data

Lesson 3

Secure Shell (SSH)

In this lesson we drill into how to use a secure shell in Linux

Secure Shell (SSH)

If you connect over an unencrypted network to a remote server network, sniffers can view the data being sent and received.

Secure Shell (SSH) uses public/private key pairs (asymmetric) for its encryption.

Linux uses OpenSSH for secure shell services

Ssh Command

The ssh command provides a secure encrypted connection between two hosts over an insecure network. This connection can also be used for terminal access, file transfers, and for tunneling other applications. Graphical X11 applications can also be run securely over SSH from a remote location.

Example Usage:

- ssh aspenvps.servers.com

Options:

- -C: use data compression
- -l: use different user
- -p: specify remote port

Remotely Executing Commands with SSH

The ssh command is often also used to remotely execute commands on the remote machine without logging in to a shell prompt.

The syntax is:

- ssh hostname command

For example, to execute the command:

- ls /home/aspeno

On host aspenvps.servers.com, type the following command at a shell prompt:

- ssh aspenvps.servers.com ls /home/aspeno

After authenticating to the remote server, the contents of the remote directory will be displayed, and you will return to your local shell prompt.

Lesson 3 Review

SSH uses PKI to encrypt network communications

You can get a remote shell via SSH

You can execute a remote command without a shell prompt and return the output

Lesson 4

Virtual Private Network (VPN)

In this lesson, we look at how we can setup a Virtual Private Network

How VPNs Work

Over Public Network

A VPN establishes a secure encrypted connection between two systems on separate networks with a public network between them

Separate Private Network

The encrypted connection acts as a separate private network

VPN Software

There are many different VPN packages available on Linux, such as OpenVPN

OpenVPN

OpenVPN provides the software for many configurations

Architecture

Create secure point-to-point or site-to-site connections in routed or bridged configurations

Client or Server

OpenVPN provides both client and server applications

Peer Authentication

Users pre-shared secret keys, certificates or username/password

Multiclient-server Configuration

Uses SSL/TLS for authentication

Lesson 4 Review

VPNs can connect two or more networks securely

VPNs can run across public networks

OpenVPN is the standard VPN for Linux