

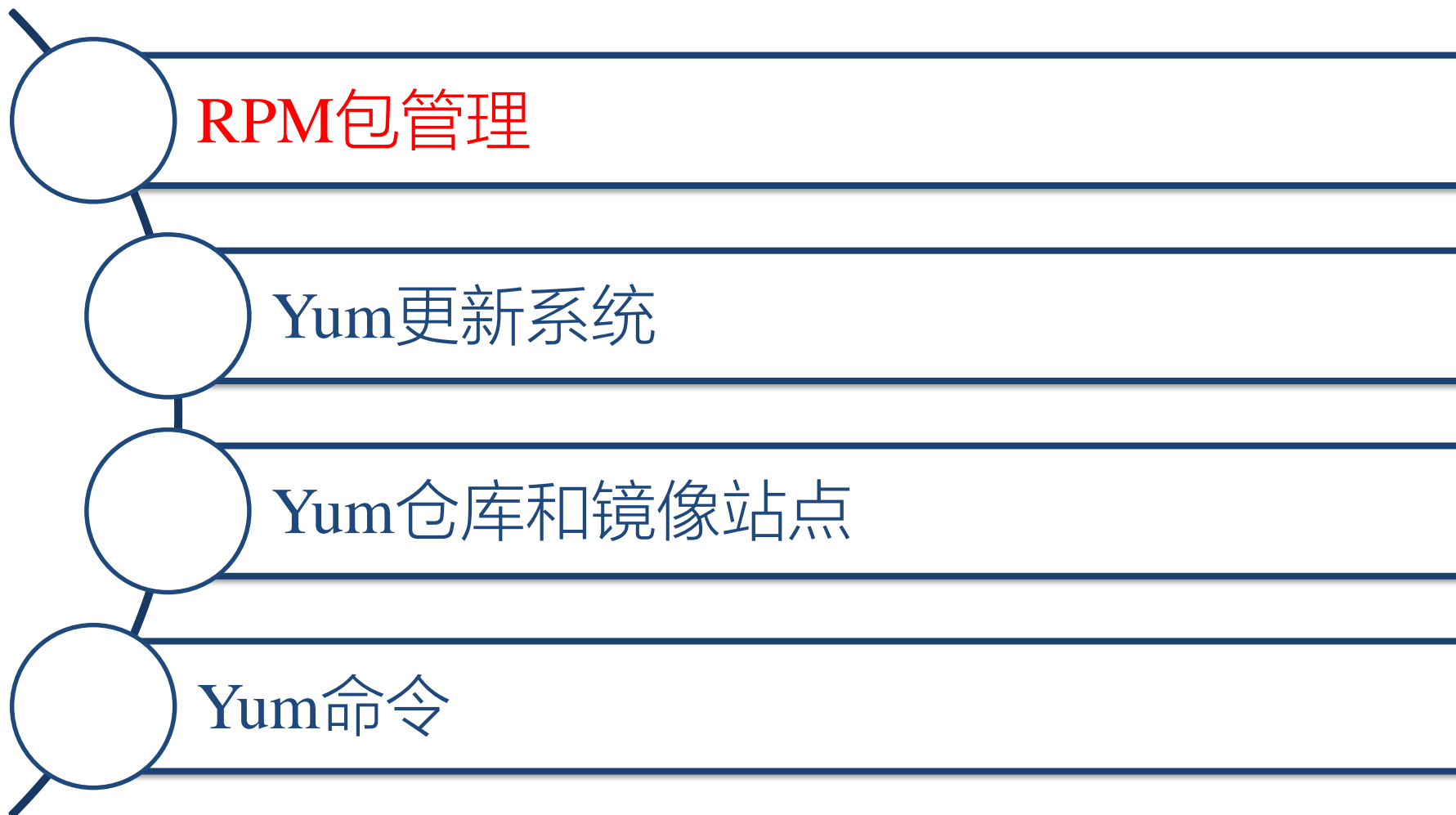
第07章 软件管理

讲师：武永亮

课程目标

- 了解RPM包管理
- 掌握Yum命令

课程内容



RPM概述

- RPM 最早是由 Red Hat 公司提出的软件包管理标准，最初的全称是 Red Hat Package Manager。
- 后来随着版本的升级又融入了许多其他的优秀特性，成为了Linux中公认的软件包管理标准。
- 被许多Linux发行使用，如：RHEL/CentOS/Fedora, SLES/openSUSE 等。
- 如今RPM是RPM Package Manager的缩写，由RPM社区（<http://www.rpm.org/>）负责维护。

RPM的优点

- 易于安装、升级便利
- 丰富的软件包查询功能
- 软件包内容校验功能
- 支持多种硬件平台

RPM的五大功能

- 安装——将软件从包中解出来，并安装到硬盘。
- 卸载——将软件从硬盘清除。
- 升级——替换软件的旧版本。
- 查询——查询软件包的信息。
- 验证——检验系统中的软件与包中软件的区别。

RPM 组件

- 本地数据库
- rpm及其相关的可执行文件
- RPM 前端工具，如 Yum
- 软件包文件

RPM包的名称格式

- name : 软件的名称
- version : 软件的版本号
- type : 包的类型
 - ✓ i[3456]86 : 在Intel x86计算机平台上编译的
 - ✓ x86_64 : 在Intel x86_64计算机平台上编译的
 - ✓ sparc/ alpha : 在sparc / alpha计算机平台上编译的
 - ✓ src : 软件源代码
- rpm : 文件扩展名

name-version.type.rpm

如: **zsh-3.0.5-15.{i386,x86_64,src}.rpm**

获得RPM包

- 从发行套件的CD中查找
- 从软件的主站点查找下载
- 从<http://www.rpmfind.net>查找下载
- 从<http://atrpms.net/>查找下载
- 从<http://rpm.pbone.net/>查找下载

安装、升级和删除软件

- 输出选项：

- ✓ -v：安装时显示软件名称

- ✓ -h：使用 “#”显示进度

- rpmfile 的URL支持

- ✓ ftp://

- ✓ http://

安装：	rpm	-i --install	<rpmfile> ...
-----	-----	---------------------	---------------

升级：	rpm	-U --upgrade	<rpmfile> ...
-----	-----	---------------------	---------------

刷新：	rpm	-F --freshen	<rpmfile> ...
-----	-----	---------------------	---------------

删除：	rpm	-e --erase	<package> ...
-----	-----	-------------------	---------------

RPM的基本查询

- 查询已安装的所有软件包
 - ✓ rpm -qa
- 查询软件包是否安装并查看软件包的版本
 - ✓ rpm -q <package_name>
- 查询软件包信息
 - ✓ rpm -qi <package_name>
 - ✓ rpm -qip <package_file_path_name>
- 查询软件包中所有文件的名称
 - ✓ rpm -ql <package_name>
 - ✓ rpm -qlp <package_file_path_name>
- 查询磁盘上的文件是从何软件包安装的
 - ✓ rpm -qf <path_name>

RPM的更多查询

- 查询依赖于一个已安装软件包的所有RPM包
 - ✓ `rpm -q --whatrequires <package-name>`
- 查询一个已安装软件包的依赖要求
 - ✓ `rpm -q --requires <package-name>`
- 查询一个已安装软件包的安装、删除脚本
 - ✓ `rpm -q --scripts <package-name>`
- 查询与一个已安装软件包相冲突的RPM包
 - ✓ `rpm -q --conflicts <package-name>`
- 查询一个已安装软件包的变更日志
 - ✓ `rpm -q --changelog <package-name>`

RPM校验

- 校验有已安装的所有软件包
 - ✓ rpm -Va
- 校验指定的软件包
 - ✓ rpm -V <package_name>
- 校验指定的RPM包文件
 - ✓ rpm -Vp <package_file_path_name>
- 验证包含指定文件的软件包
 - ✓ rpm -Vf <path_name>

RPM包的公钥和签名

- 导入RPM包的公钥

- ✓ 格式：rpm --import <公钥文件名>

- ✓ 例如

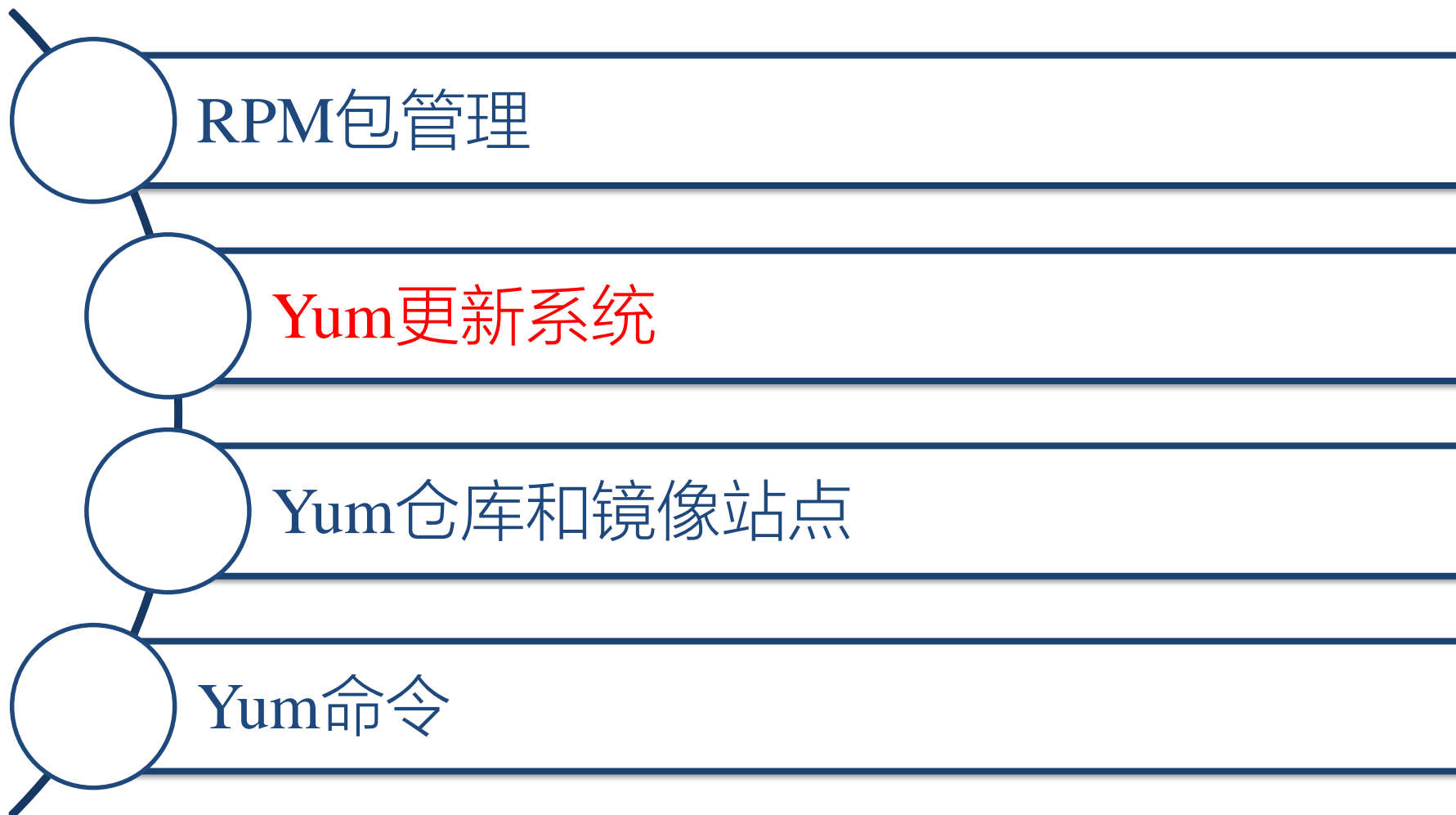
- ✓ # rpm --import /etc/pki/rpm-gpg/RPM-GPG-*

- ✓ # rpm --import <http://apt.sw.be/RPM-GPG-KEY.dag.txt>

- 检查指定RPM包的数字签名

- ✓ rpm -K <rpmfile>

课程内容



软件包管理与系统更新

- 使用软件更新系统的目的
 - ✓ 为了解决安装RPM时的依赖性问题
- 常见的基于RPM的更新系统
 - ✓ Red Hat Network —— Red Hat 的企业级更新系统
 - ✓ Yum —— Fedora, CentOS
 - ✓ zypp —— openSUSE
 - ✓ urpmi —— Mandriva
 - ✓ APT-RPM —— PCLinuxOS , ALT Linux

其他Linux发行的更新软件

- apt —— Debian, Ubuntu, LinuxMint
- apk —— Alpine
- slackpkg —— Slackware
- emerge —— Gentoo
- pacman —— Arch
- conary —— rPath, Foresight

Yum简介

- Yum 是 Yellow dog Updater, Modified 的简称，用 python 写成。
- Yum 的宗旨是自动化地升级，安装/移除rpm包，收集rpm包的相关信息，检查依赖性并自动提示用户解决。
- Yum 是 rpm 的前端程序，RHEL 的 up2date 的替代工具。
- Yum 的关键之处是要有可靠的 repository（软件仓库）
 - ✓可以是 http 或 ftp 站点，也可以是本地软件池
 - ✓包含rpm 包的各种信息（包括描述，功能，提供的文件，依赖性等）
 - ✓Yum 正是由于对收集的这些 header并加以分析，才能自动化地完成安装/更新/删除等任务

Yum 的特点

- 便于管理大量系统的更新问题
 - ✓ 自动解决包的倚赖性问题能更方便的 添加/删除/更新 RPM包
- 可以同时配置多个资源库（ Repository ）
 - ✓ 可以在多个库之间定位软件包
- 简洁的配置文件
 - ✓ /etc/Yum.conf 和 /etc/Yum.repos.d/*.repo
- 保持与RPM数据库的一致性
- 有一个比较详细的log，可以查看何时升级安装了什么软件包等

Yum组件

- Yum命令

- ✓通过Yum命令使用Yum提供的众多功能。
- ✓由名为 “Yum”软件包提供（默认已安装）。
- ✓Yum软件的主页为<http://linux.duke.edu/Yum/>。

- Yum插件

- ✓由官方或第三方开发的Yum插件用于扩展Yum的功能。
- ✓通常由以名为 “Yum -<pluginname>”的软件包提供。

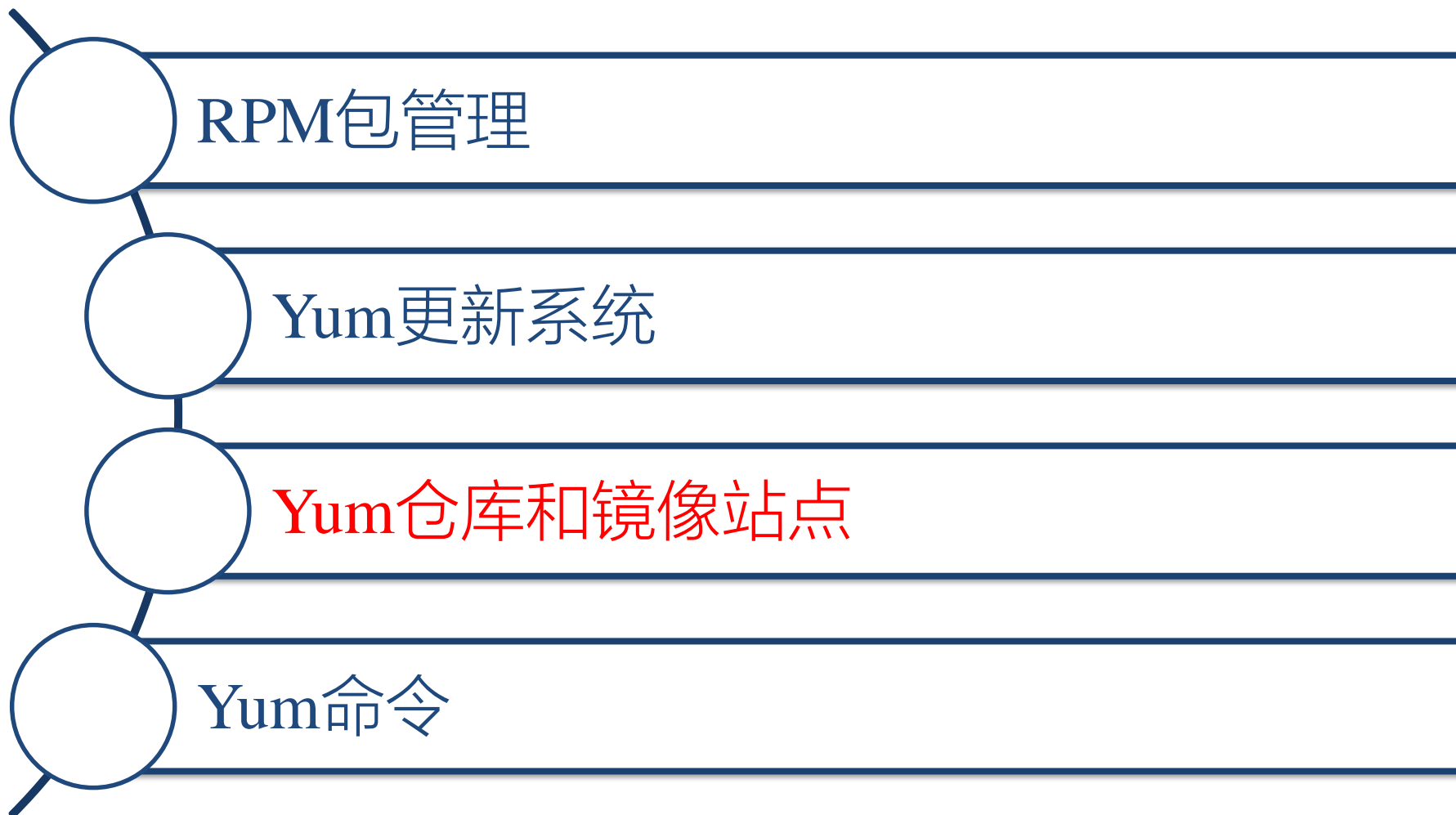
- Yum仓库

- Yum缓存

常用的Yum插件

- Yum-priorities : 设置多个仓库的使用优先级别
- Yum-versionlock : 用于锁定某软件的版本，以免更新
- Yum-changelog : 查看包更新前后的改变
- Yum-aliases : 为Yum命令使用别名
- Yum-security : 为Yum提供安全过滤器

课程内容



Yum仓库

- Yum仓库（ repository ）亦称 “更新源” 。
- 一个Yum软件仓库就是一个包含了仓库数据的存放众多RPM文件的目录。
- Yum仓库数据通常存放在名为 “repodata”的子目录中。
- Yum客户通过访问Yum仓库数据进行分析并完成查询、安装、更新等操作。
 - ✓ Yum客户可以使用http://、 ftp:// 或file://（ 本地文件 ）协议访问Yum仓库。
 - ✓ Yum客户可以使用官方和第三方提供的众多Yum仓库更新系统。
- createrepo、 Yum-utils等软件包（ 默认未安装 ）中提供了Yum仓库管理工具。

CentOS 的镜像站点

- CentOS 的 Yum仓库 位于 CentOS 的镜像站点。
- 用 Yum 命令可以通过 FTP 或 HTTP 访问远程 Yum仓库。
- 镜像站点的第一级目录是发行版本号，如 3、4、5、6 等。
- CentOS 镜像站点的版本号为6的Yum仓库。
 - ✓ os/：发行版（distributions）的base仓库
 - ✓ updates/：updates 仓库
 - ✓ SCL/：SCL(The Software Collections)仓库
 - ✓ centosplus/：centosplus 仓库
 - ✓ extras/：extras 仓库
 - ✓ fasttrack/：fasttrack 仓库
 - ✓ isos/：本目录包含发行版的 CD/ DVD isos 下载文件

<http://wiki.centos.org/AdditionalResources/Repositories>

CentOS 仓库的目录结构

- |-- i386 # Intel 32位平台目录
- | |-- Packages/ # Intel 32位平台的RPMS目录
- | | |-- *.i386.rpm # 在Intel 32位平台上编译的包文件
- | | `-- *.centos.noarch.rpm # 与平台无关的已编译的包文件
- | `-- repodata/ # Intel 32位平台的索引文件
- `-- x86_64 # 64位平台目录
- |-- Packages/ # 64位平台的RPMS目录
- | |-- *.x86_64.rpm # 在64位平台上编译的包文件
- | `-- *.centos.noarch.rpm # 与平台无关的已编译的包文件
- `-- repodata/ # 64位平台的索引文件

Yum的配置

Yum主配置文件/etc/Yum.conf

- [main]
- cachedir=/var/cache/Yum # 指定Yum缓存目录
- keepcache=0 # 是否保持缓存（包括仓库数据和RPM），1保存，0不保存
- debuglevel=2 # 设置日志记录等级(0-10)，数值越高记录的信息越多
- logfile=/var/log/Yum.log # 设置日志文件路径
- distroverpkg=redhat-release # 指定发行版本的软件包名称
- tolerant=1 # 允许Yum在出现错误时继续运行，比如不需要更新的程序包
- exactarch=1 # 更新时不允许更新不同版本的RPM包
- obsoletes=1 # 相当于upgrade，允许更新陈旧的RPM包
- gpgcheck=1 # 校验软件包的GPG签名
- plugins=1 # 默认开启Yum的插件使用
- metadata_expire=1h # 设置仓库数据的失效时间为1小时
- installonly_limit = 5 # 允许保留多少个内核包
- reposdir = /etc/Yum.repos.d # 指定仓库配置文件的目录，此为默认值

Yum的仓库配置语法

- [repositoryid]
- name=name for this repository
- baseurl=url://server1/path/to/repository/
 - url://server2/path/to/repository/
 - url://server3/path/to/repository/
- mirrorlist=url://path/to/mirrorlist/repository/
- enabled=0/1
- gpgcheck=0/1
- gpgkey=A URL pointing to the GPG key file

设置网络更新源/etc/Yum.repos.d/*.repo

● 网络更新源

- ✓ 默认配置文件：CentOS-Base.repo
- ✓ 下载使用国内的镜像站点提供的仓库配置文件
 - <http://mirrors.sohu.com/help/CentOS-Base-sohu.repo>
 - <http://mirrors.163.com/.help/CentOS-Base-163.repo>
 - <http://centos.ustc.edu.cn/CentOS-Base.repo.6>

● 本地更新源

- ✓ 默认配置文件：CentOS-Media.repo

使用非官方软件仓库

为什么使用非官方仓库

- 官方仓库是指RedHat/CentOS提供的仓库
- 非官方仓库是指官方仓库之外的由其他社区或某软件制作者提供的仓库。
- 使用非官方仓库的目的
 - ✓ 安装官方仓库中不提供的软件包
 - ✓ 安装比官方仓库中版本更新的软件包
- 应该选择使用何种非官方仓库
 - ✓ 知名的非官方仓库
 - ✓ 具有GPG签名的非官方仓库

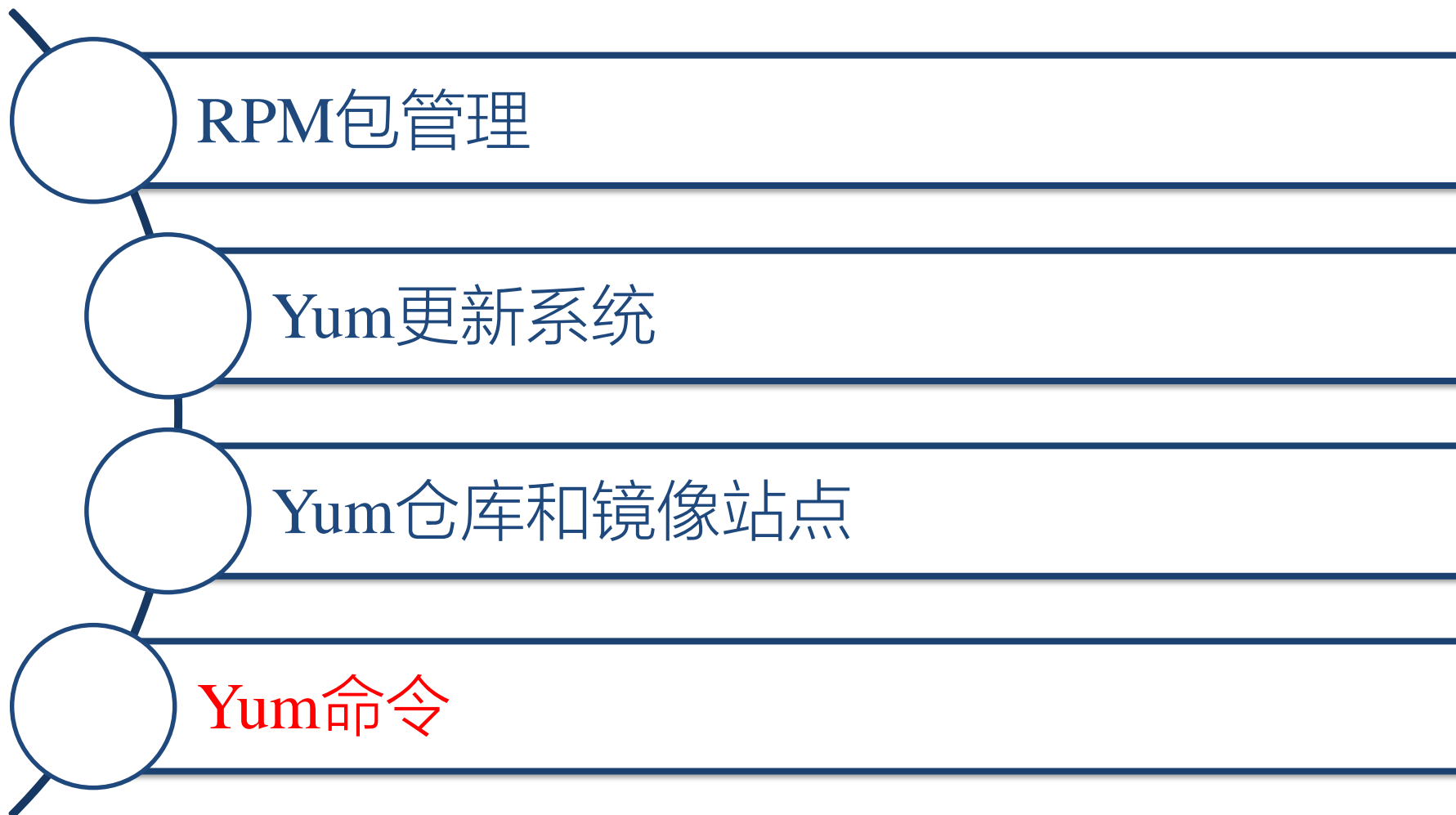
常用的非官方仓库

仓库名	URL
epel	http://fedoraproject.org/wiki/EPEL
rpmforge	http://rpmforge.net/
remi	http://rpms.famillecollet.com/
rpmfusion	http://rpmfusion.org/
atrpms	http://atrpms.net/
webmin	http://www.webmin.com.cn/rpm.html
openvz	http://wiki.openvz.org/Yum

使用非官方仓库有两种方法

- 提供仓库 "release" RPM包的非官方仓库
 - ✓ 下载非官方仓库的 "release" RPM包
 - ✓ 导入仓库的 RPM 公钥文件并验证 "release" RPM包
 - ✓ 使用 rpm 命令安装非官方仓库的 "release" RPM包
- 未提供仓库 "release" RPM包的非官方仓库
 - ✓ 进入 /etc/Yum.repos.d 目录
 - ✓ 下载或直接编辑 ".repo"文件
 - ✓ 导入仓库的 RPM 公钥

课程内容



软件管理

- 使用统一的命令管理软件的目的
 - ✓ 为了解决安装RPM时的依赖性问题
- 常见的基于RPM的系统
 - ✓ Red Hat Network —— Red Hat 的企业级更新系统
 - ✓ Yum —— Fedora, CentOS
 - ✓ APT-RPM —— PCLinuxOS , ALT Linux
 - ✓ zypp —— openSUSE
 - ✓ urpmi —— Mandriva

其他Linux发行的更新软件

- apt —— Debian, Ubuntu, LinuxMint
- apk —— Alpine
- slackpkg —— Slackware
- emerge —— Gentoo
- pacman —— Arch
- conary —— rPath, Foresight

Yum简介

- Yum (全称为 Yellow dog Updater, Modified) 是Fedora和RedHat以及CentOS中的Shell前端软件包管理器。基于RPM包管理，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包，无须繁琐地一次次下载、安装。
- Yum 的关键之处是要有可靠的 repository (软件仓库)
 - ✓ 可以是 http 或 ftp 站点，也可以是本地软件池
 - ✓ 包含rpm 包的各种信息 (包括描述，功能，提供的文件，依赖性 等)
 - ✓ Yum 正是由于对收集的这些 header并加以分析，才能自动化地完成安装/更新/删除等任务

Yum命令语法

- yum是Yum系统的字符界面管理工具
 - ✓ yum [全局参数] 命令 [命令参数]
- 常用的全局参数
 - ✓ -y：对Yum命令的提问回答“是（yes）”
 - ✓ -C：只利用本地缓存，不从远程仓库下载文件
 - ✓ --enablerepo=REPO：临时启用指定的名为REPO的仓库
 - ✓ --disablerepo=REPO：临时禁用指定的名为REPO的仓库
 - ✓ --installroot=PATH：指定安装软件时的根目录，主要用于为chroot环境安装软件

Yum命令语法

- 搜索软件 : yum search tree
- 安装软件 : yum install tree
- 删除软件 : yum remove tree
- 升级软件 : yum update tree
- 查询信息 : yum info tree
- 显示软件包依赖关系 : yum deplist tree
- 检查可更新的所有软件包 : yum check-update

Yum的练习

- 下载安装tree相关包
 - ✓ yum install tree
- 下载安装Vim相关包
 - ✓ yum install vim-enhanced
- 下载安装neon相关包
 - ✓ yum install neon
- 下载安装Memcached相关包
 - ✓ yum install memcached

Yum的练习

- 下载安装MySQL相关包

- ✓ yum install mysql

- 下载安装Apache相关包

- ✓ yum install httpd

- 下载安装Nginx相关包

- ✓ rpm -Uvh <http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm>

- ✓ yum install nginx

- 下载安装PHP相关包

- ✓ yum install PHP

Yum 仓库管理

Yum仓库管理工具

- 软件包createrepo
 - ✓ 提供了 createrepo命令用于生成Yum仓库
- 软件包Yum-utils主要提供了如下常用工具
 - ✓ Yumdownloader：从Yum仓库（包括SRPMs）下载RPM文件。
 - ✓ reposync：使用Yum配置检索Yum远程仓库并同步到本地目录。
 - ✓ verifytree：校验本地Yum仓库的一致性。
 - ✓ Yum-complete-transaction：查找并处理Yum完整性。

本地仓库创建过程

- 创建存放RPM包的目录
- 在RPM包的目录中准备RPM包文件：
 - ✓1) 从安装光盘获得
 - ✓2) 通过wget、lftp等工具从远程下载
 - ✓3) 通过 Yumdownloader 工具从远程下载
 - Yumdownloader --resolve #可以同时下载被依赖的RPM包
 - Yumdownloader --source #可以下载SRPM的RPM包
 - ✓4) 可以使用 rpmbuild命令本地编译
- 使用createrepo命令生成本地仓库

createrepo 命令

- 命令格式

- ✓ createrepo [选项] 包目录

- 常用选项

- ✓ -g, --groupfile <filename> : 指定Yum组操作所需的XML文件

- ✓ -d, --database : 生成sqlite 数据库文件

- ✓ --update : 更新仓库的元数据文件

- ✓ -q, --quiet : 不显示操作过程

- ✓ -v, --verbose : 显示完整的操作过程

- ✓ -h, --help : 显示帮助信息

课程总结



本章思考题

- 什么是RPM？为什么使用RPM？RPM具有什么功能？
- 举例说明使用RPM命令安装、升级、删除、查询、校验软件包的方法。
- 为何使用Yum？Yum常用命令及参数有哪些？
- 如何创建本地仓库？
- 镜像远程仓库可以使用哪些命令工具？

本章实验

- 学会使用RPM命令。
- 学会使用Yum进行系统更新。
- 学会配置Yum仓库配置文件。

THANK YOU!