Linux服务器构建与运维管理

从基础到实战（基于 openEuler）

第4章：使用Apache HTTP Server实现网站服务

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

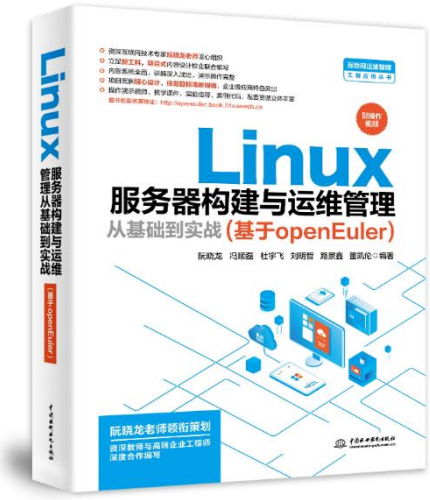
https://internet.hactcm.edu.cn
http://www.51xueweb.cn

河南中医药大学信息技术学院互联网技术教学团队
河南中医药大学医疗健康信息工程技术研究所

2024.9

提纲

- Apache Http Server
  - Web 服务器（网站服务器）
  - Apache Http Server
- 发布静态网站
  - 安装、发布网站、测试
- 建设内容网站服务
  - 安装并实现 LAMP 环境
  - 使用WordPress建设内容网站
- 查看Apache的服务状态
  - Apache Module
  - mod_status、mod_info
- 提升Apache的安全性



1. Apache Http Server

- Web服务器是一种驻留于因特网上的计算机程序，实现两个作用：
  - 放置网站程序，让用户通过浏览器访问
  - 放置数据文件，让用户下载

- Web服务器端对客户端提供的网站程序有两种类型：
  - 静态文档：html、JS
  - 动态文档：PHP、ASP.net、JSP、Ruby、Python

- Web服务器使用HTTP进行通信，因此Web服务器亦称为HTTP服务器。

1. Apache Http Server

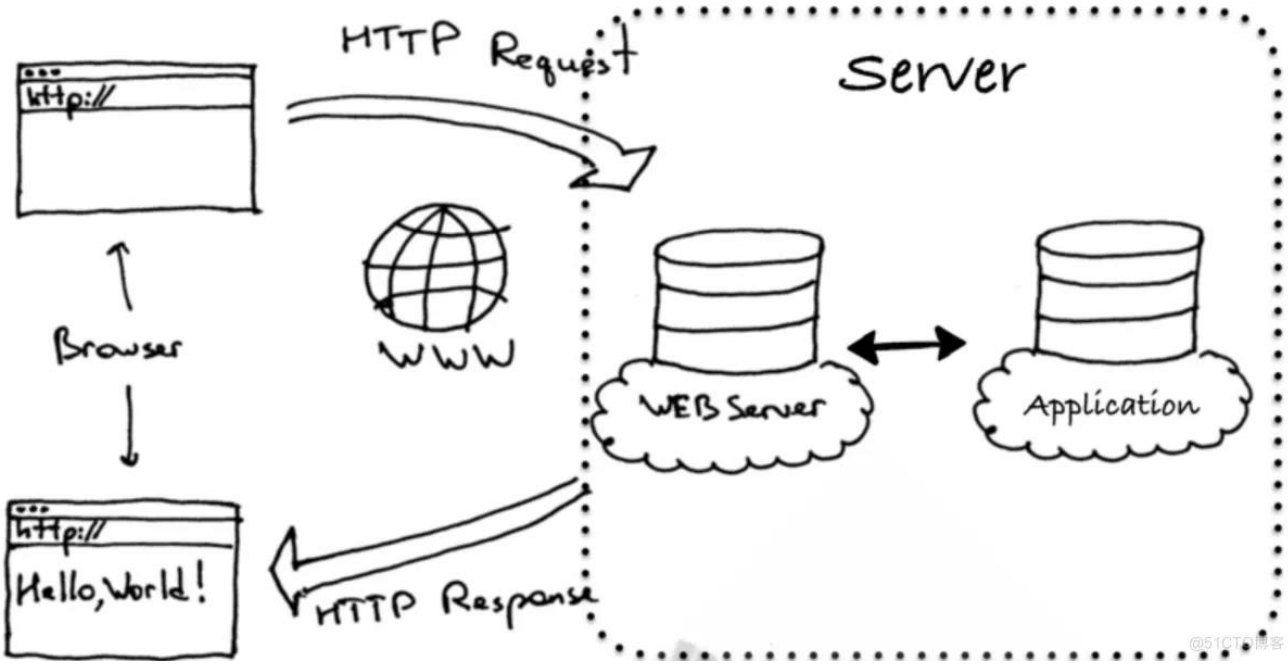
- 由于不同的Web服务器对HTTP请求的处理方式并不完全相同，进行相应的处理时采用的资源分配策略和调度的方式也各有差异。
- 适应为各种不同功能、不同环境的Web服务器不断出现，目前常用的Web服务器是Apache Http Server、IIS、Tomcat、Nginx等。

- Apache Http Server是Linux平台中应用最广泛的Web服务器。

1. Apache Http Server

- Web服务器处理分为四步：连接、请求、应答、关闭。
  - 连接过程：
    - 根据输入的URL地址，Web客户端（如浏览器）连接到相应的Web服务器上。
  - 请求过程：
    - Web客户端运用socket向其服务器提出各种请求。
  - 应答过程：
    - Web服务器接收到请求后实施任务处理，使用HTTP协议把任务处理的结果传输到Web客户端。
    - Web客户端获取数据后展示给用户。
  - 关闭连接：
    - 当应答过程完成以后，Web服务器和Web客户端之间断开连接。

Web 服务器的基本工作原理



1. Apache Http Server

- Apache Http Server是最常用的开源网站服务器软件之一。
  - 支持UNIX、Linux、Windows等操作系统。
  - Apache HTTP Server 通常被简称为Apache。
  - Apache目前最新版本为 2.4.62。
  - 本课程在openEuler上使用的版本为2.4.58。
  - Apache HTTP Server 官网为 https://www.apache.org

Apache HTTP Server (a project managed by the Apache HTTP Server Committee)

The Apache HTTP Server is an open-source HTTP server for modern operating systems including UNIX, Microsoft Windows, Mac OS/X and Netware. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services observing the current HTTP standards. Apache has been the most popular web server on the Internet since April of 1996.

Project base data:

- **Short description:** The Apache HTTP Server application 'httpd'.
- **Category:** network-server http httpd-module
- **Website:** http://httpd.apache.org/
- **Project status:** Active
- **Implemented standards**
  - IETF RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1
  - IETF RFC 2617: HTTP Authentication: Basic and Digest Access Authentication
  - IETF RFC 2518: HTTP Extensions for Distributed Authoring -- WEBDAV
  - IETF RFC 3253: Versioning Extensions to WebDAV
  - IETF RFC 3986: Uniform Resource Identifier (URI): Generic Syntax
- **Project data file:** DOAP RDF Source (generated json)

Development:

- **Programming language:** C
- **Bug-tracking:** http://httpd.apache.org/bug_report.html
- **Mailing list(s):** http://httpd.apache.org/lists.html
- **Subversion repository:** http://svn.apache.org/repos/asf/httpd/httpd/

Releases (from DOAP):

- **Download:** http://httpd.apache.org/download.cgi
- **2.4.62** (2024-07-17): Recommended current 2.4 release

APACHE HTTP SERVER PROJECT

COMMUNITY CODE Denver, Colorado October 7-10, 2024

Essentials

- Download!
- About
- License
- FAQ
- Security Reports

Source Repositories

- General Information
- Trunk
- 2.4

Documentation

- Version 2.4
- Trunk (dev)
- Wiki

Get Involved

- Mailing Lists
- Bug Reports
- Developer Info
- User Support

Subprojects

- Docs
- Test
- Flood
- libapreq
- Modules
- mod_fcgid
- mod_ftp

Related Projects

- Apache Traffic Server
- Apache Traffic Control
- Tomcat
- APR
- mod_perl

Miscellaneous

- Contributors
- Thanks!
- Sponsorship

The Number One HTTP Server On The Internet

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. It has celebrated its 25th birthday as a project in February 2020.

The Apache HTTP Server is a project of The Apache Software Foundation.

Apache httpd 2.4.62 Released 2024-07-17

The Apache Software Foundation and the Apache HTTP Server Project are pleased to announce the release of version 2.4.62 of the Apache HTTP Server ("httpd").

This latest release from the 2.4.x stable branch represents the best available version of Apache HTTP Server.

Apache HTTP Server version 2.4.43 or newer is required in order to operate a TLS 1.3 web server with OpenSSL 1.1.1.

Download | ChangeLog for 2.4.62 | Complete ChangeLog for 2.4 | New Features in httpd 2.4

Apache httpd 2.2 End-of-Life 2018-01-01

As previously announced, the Apache HTTP Server Project has discontinued all development and patch review of the 2.2.x series of releases.

The Apache HTTP Server Project had long committed to provide maintenance releases of the 2.2.x flavor through June of 2017. The final release 2.2.34 was published in July 2017, and no further evaluation of bug reports or security risks will be considered or published for 2.2.x releases.

Want to try out the Apache HTTP Server?

Great! We have updated our download page in an effort to better utilize our mirrors. We hope that by making it easier to use our mirrors, we will be able to provide a better download experience.

Please ensure that you verify your downloads using PGP or MD5 signatures.

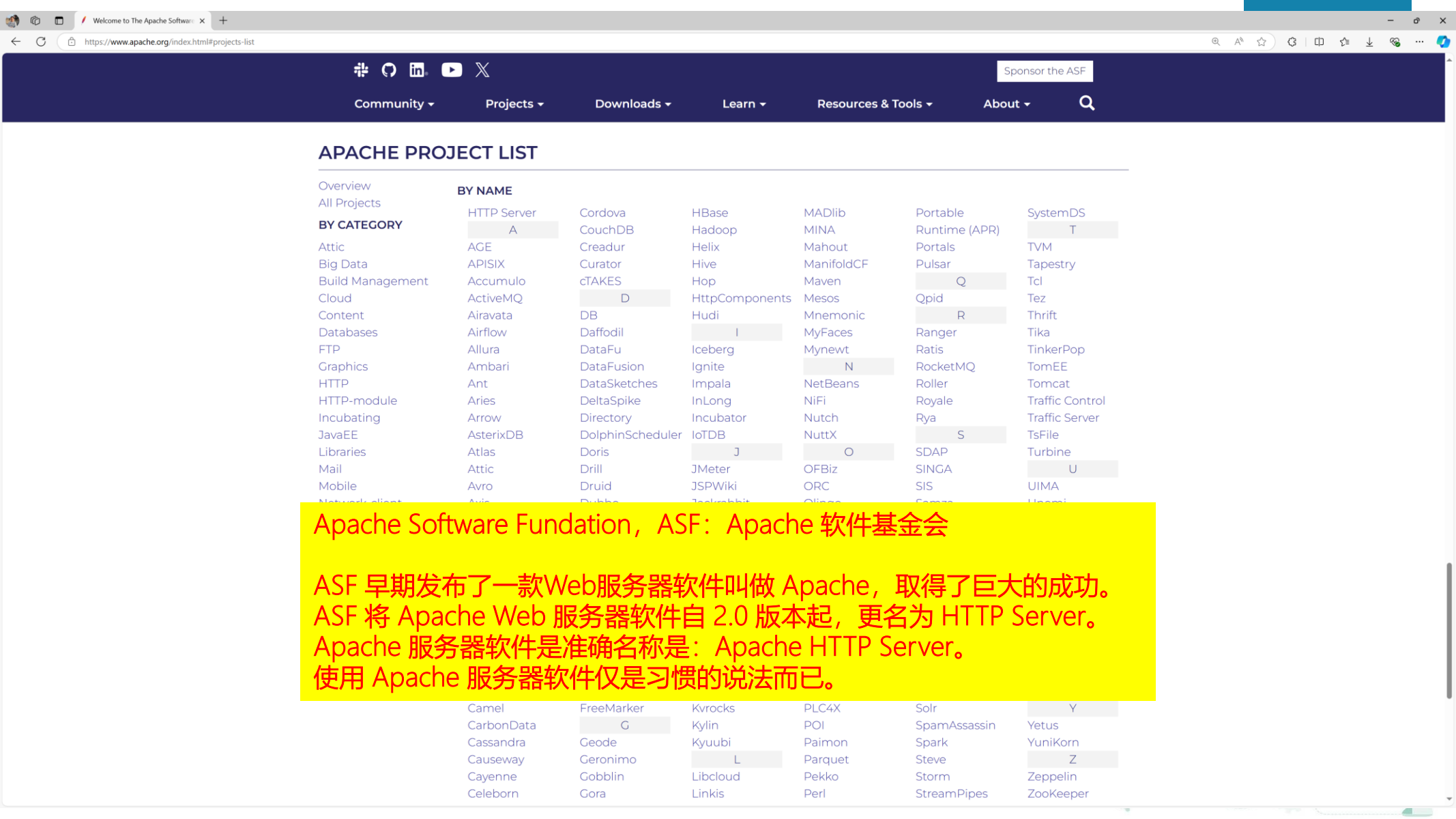
Want to contribute to the Apache HTTP Server?

Awesome! Have a look at our current 'Help Wanted' listings then:

Tasks the Apache HTTP Server Project would like help with:

Title	Languages	Difficulty	Created
Rewrite mod_ldap cache handling code	c	Intermediate	2018/5/29 01:23:20
Learn and work on httpd's internals	c	Advanced	2018/1/3 01:19:40
Websocket documentation	english,xml,lua,c	Intermediate	2016/2/20 01:31:35
Improve the Request Processing guide	xml,html,c	Advanced	2016/2/14 19:16:11
Traffic shaping with mod_lua	lua,c	Advanced	2016/2/14 17:09:50
Make a better guide on request hooks	xml,html,c	Advanced	2016/2/8 17:46:07
Write a better cache guide	xml,html	Expert	2016/2/7 05:30:59

Powered by **Help Wanted** - a task directory for Apache projects



Apache Software Fundation，ASF：Apache 软件基金会

ASF 早期发布了一款Web服务器软件叫做 Apache，取得了巨大的成功。
ASF 将 Apache Web 服务器软件自 2.0 版本起，更名为 HTTP Server。
Apache 服务器软件是准确名称是：Apache HTTP Server。
使用 Apache 服务器软件仅是习惯的说法而已。

1. Apache Http Server

- Apache HTTP Server 的主要特性如下。
  - 支持最新的 HTTP 协议和多种方式的 HTTP 认证。
  - 支持基于文件的配置。
  - 支持基于 IP 和域名的虚拟网站配置。
  - 支持通用网关接口，支持 PHP、FastCGI、Perl、JavaServlets 等。
  - 支持服务器状态监控。
  - 支持服务器日志记录和日志格式自定义设置。
  - 支持服务器端包含指令（SSI）。
  - 支持安全 Socket 层（SSL）。
  - 集成代理服务器模块。


Apache HTTP 服务器 2.4 文档 - A
https://httpd.apache.org/docs/2.4/
APACHE
HTTP SERVER PROJECT
Apache HTTP 服务器版本 2.4
模块 | 指令 | 常见问题 | 术语 | 网站导航
Apache > HTTP 服务器 > 文档
Apache HTTP 服务器 2.4 文档
可用语言: da | de | en | es | fr | ja | ko | pt-br | ru | tr | zh-cn
Google 搜索
发行说明
Apache 2.3/2.4 的新特性
Apache 2.1/2.2 的新特性
Apache 2.0 的新特性
从 2.2 升级到 2.4
Apache 许可证
参考手册
编译与安装
启动
停止与重启
指令快速参考
模块
多处理模块(MPM)
过滤器
处理器
表达式解析器
服务器与支持程序
术语
用户指南
入门指南
绑定指定地址与端口
配置文件
配置片段
缓存指南
内容协商
动态共享对象(DSO)
环境变量
日志文件
从 URL 映射到文件系统
性能调谐
安全技巧
服务器全局配置
SSL/TLS 加密
执行 CGI 前的用户切换(suEXEC)
URL 改写与 mod_rewrite
虚拟主机
指引/教程
认证，授权与访问控制
访问控制
CGI 与动态内容
.htaccess 文件
服务器端插入(SSI)
用户私人网站目录(public_html)
反向代理设置指南
HTTP/2 指南
平台相关说明
Microsoft Windows
基于RPM安装包的系统 (Redhat / CentOS / Fedora)
Novell NetWare
EBCDIC 系统
其它主题
常见问题
网站导航
开发文档
帮助改进文档
其它说明
维基
可用语言: da | de | en | es | fr | ja | ko | pt-br | ru | tr | zh-cn
模块 | 指令 | 常见问题 | 术语 | 网站导航
Copyright 2024 The Apache Software Foundation.
基于 Apache License, Version 2.0 许可证.
养成阅读官方文档的习惯，是最有效的学习途径。

Binding to Addresses and Ports

Configuring Apache HTTP Server to listen on specific addresses and ports.

Overview

Related Modules	Related Directives
core	<VirtualHost>
mpm_common	Listen

When httpd starts, it binds to some port and address on the local machine and waits for incoming requests. By default, it listens to all addresses on the machine. However, it may need to be told to listen on specific ports, or only on selected addresses, or a combination of both. This is often combined with the Virtual Host feature, which determines how `httpd` responds to different IP addresses, hostnames and ports.

The `Listen` directive tells the server to accept incoming requests only on the specified port(s) or address-and-port combinations. If only a port number is specified in the `Listen` directive, the server listens to the given port on all interfaces. If an IP address is given as well as a port, the server will listen on the given port and interface. Multiple `Listen` directives may be used to specify a number of addresses and ports to listen on. The server will respond to requests from any of the listed addresses and ports.

For example, to make the server accept connections on both port 80 and port 8000, on all interfaces, use:

```
Listen 80
Listen 8000
```

To make the server accept connections on port 80 for one interface, and port 8000 on another, use

```
Listen 192.0.2.1:80
Listen 192.0.2.5:8000
```

IPv6 addresses must be enclosed in square brackets, as in the following example:

```
Listen [2001:db8::a00:20ff:fea7:ccea]:80
```

Overlapping `Listen` directives will result in a fatal error which will prevent the server from starting up.

```
(48)Address already in use: make_sock: could not bind to address [::]:80
```

See the discussion in the wiki for further troubleshooting tips.

Changing Listen configuration on restart

When httpd is restarted, special consideration must be made for changes to `Listen` directives. During a restart, httpd keeps ports bound (as in the original configuration) to avoid generating "Connection refused" errors for any new attempts to connect to the server. If changes are made to the set of `Listen` directives used which conflict with the old configuration, configuration will fail and the server will terminate.

For example, changing from configuration:

```
Listen 127.0.0.1:80
```

to the following may fail, because binding to port 80 across all addresses conflicts with binding to port 80 on just 127.0.0.1.

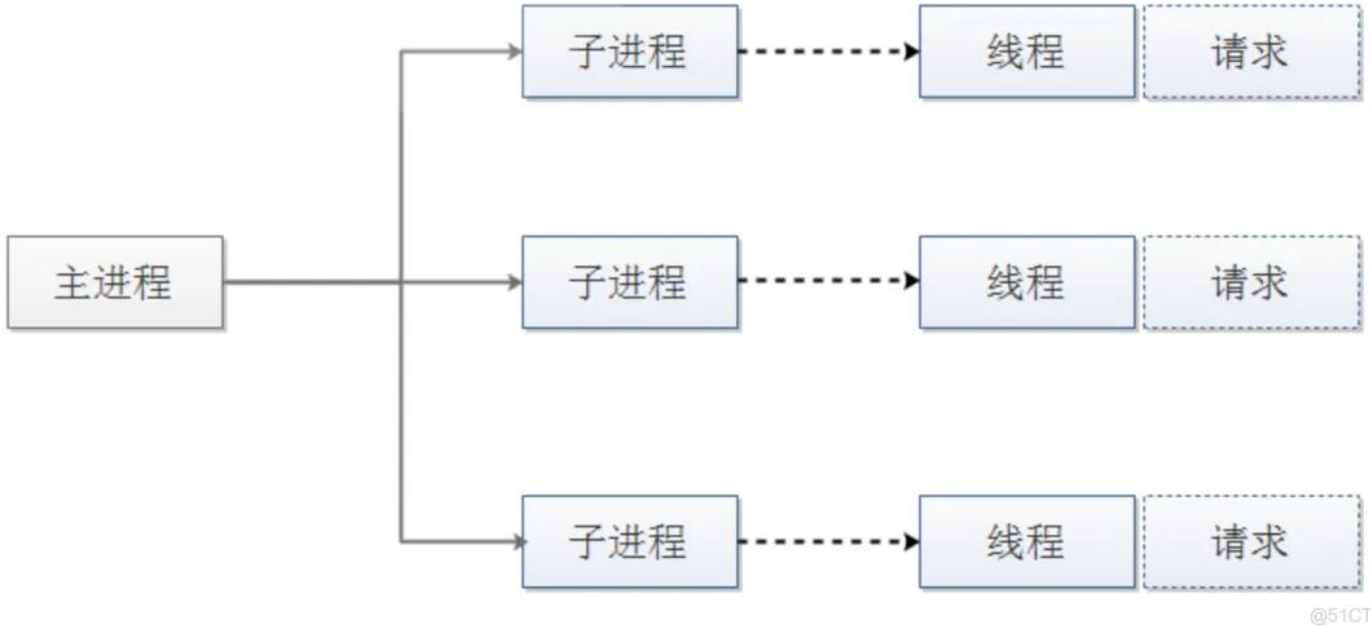
如何配置，以及配置文件的具体写法，

在官方文档中都有明确的解释。

1. Apache Http Server

- Apache HTTP Server有prefork、worker和event3种工作模式。
  - prefork工作模式。prefork是稳定模式。
    - Apache在启动之初，就预先派生一些子进程，然后等待客户端的请求进来，用于减少频繁创建和销毁进程的开销。每个子进程只有一个线程，在一个时间点内，只能处理一个请求。其缺点是它将请求放进队列中，一直等到有可用进程，请求才会被处理。
    - Prefork下有 StartServers、MinSpareServers、MaxSpareServers和MaxRequestWorkers 4个指令用于调节父进程如何产生子进程。
    - 4个指令的含义如下所示。
      1）StartServers：初始的工作进程数。
      2）MinSpareServers：空闲子进程的最小数量。
      3）MaxSpareServers：空闲子进程的最大数量。
      4）MaxRequestWorkers：最大空闲线程数。
    - 通常情况下，Apache具有很强的自我调节能力，不需要额外调整。
    - 但当需要处理的并发请求较高时，服务器可能就需要增加 MaxRequestWorkers 的值。
    - 内存较小的服务器需要减少 MaxRequestWorkers 的值以确保服务器不会崩溃。

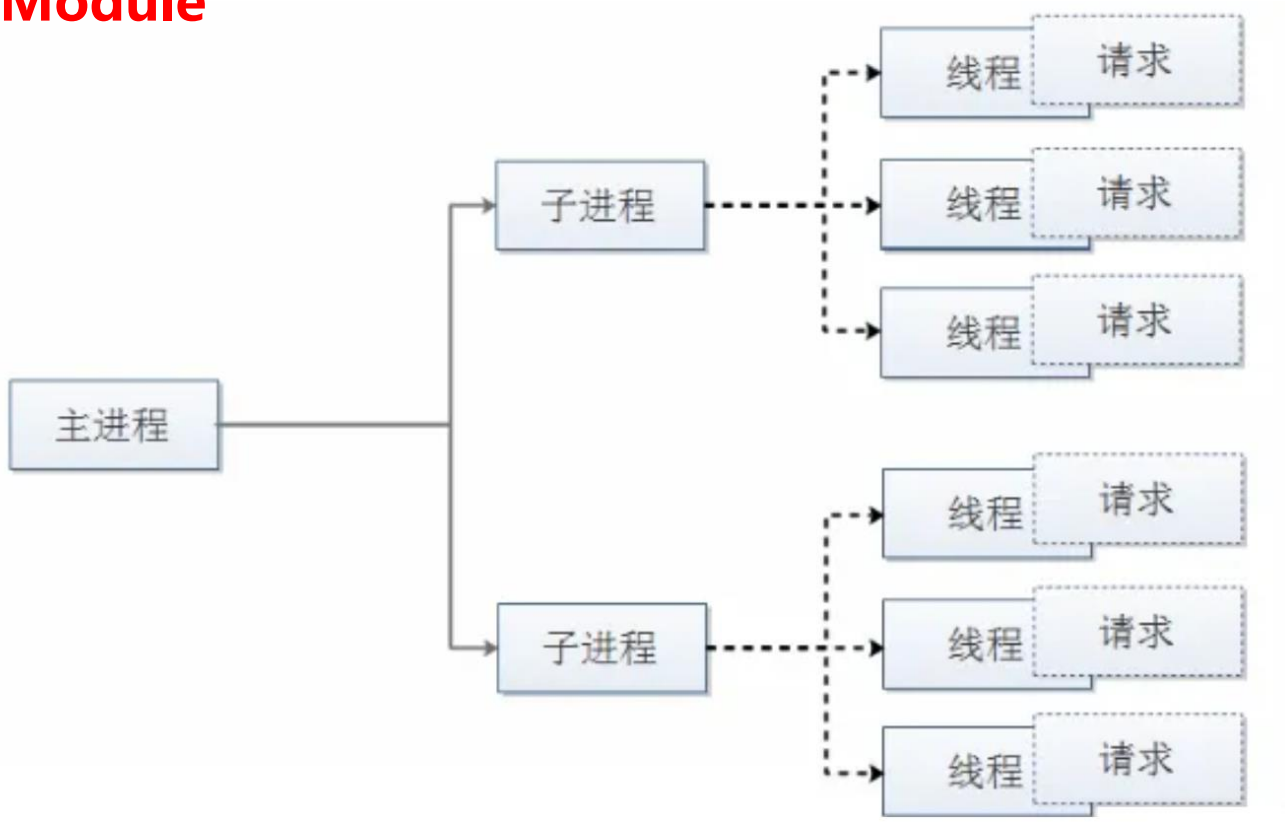
Apache Prefork Module



1. Apache Http Server

- Apache HTTP Server有prefork、worker和event 3种工作模式。
  - worker工作模式。
    - Worker模式相对于prefork模式来说，使用多进程和多线程混合模式。
    - Apache启动时预先分了几个子进程（数量比较少），每个子进程创建一些线程，同时包括一个监听线程。每个请求过来，会分配一个线程来进行服务。线程通常会共享父进程的内存空间，对内存占用会减少些，用线程处理会更轻量。
    - worker模式在高并发的情况下，比prefork模式有更多的可用进程。
    - 考虑到稳定性，worker模式不完全使用多线程，还引入多进程。如果使用单进程，在一个线程出错往往会导致父进程连同其他正常的子线程都出错。使用多个进程加多个线程的方式，即便某个线程出现异常，受影响的只有Apache的部分服务。

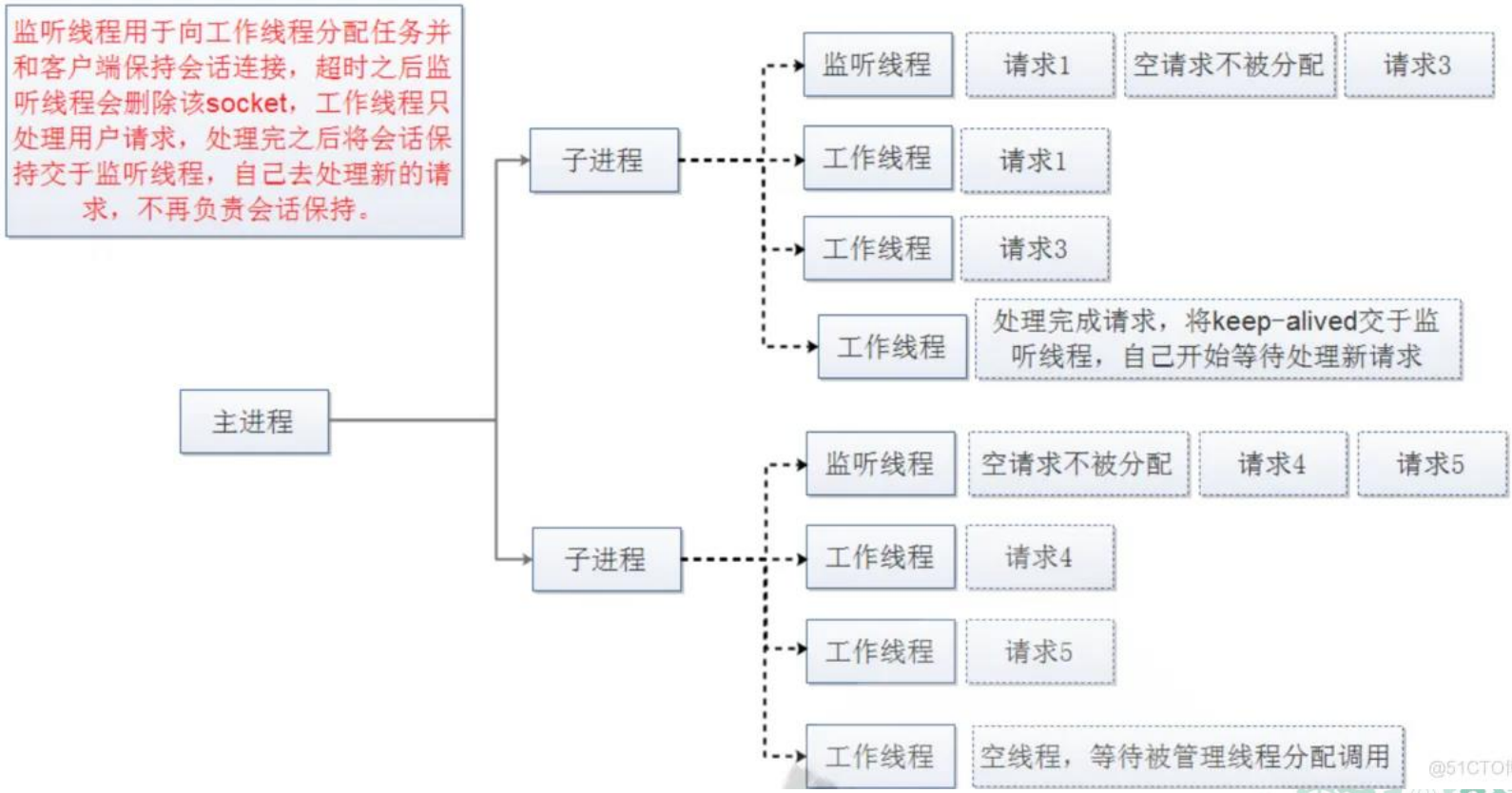
Apache Work Module



1. Apache Http Server

- Apache HTTP Server有prefork、worker 和 event 3种工作模式。
  - event工作模式。
    - event和worker模式较为相似，但event模式解决了keep-alive场景下线程长期被占用而造成的资源浪费问题。
    - event模式中，会有一个专门的线程来管理keep-alive类型的线程。
    - 当有真实请求时将请求传递给服务线程，执行完毕后释放，增强了高并发场景下的请求处理能力。

Apache Event Module



2. 发布静态网站

通过 Apche HTTP Server 发布静态网站

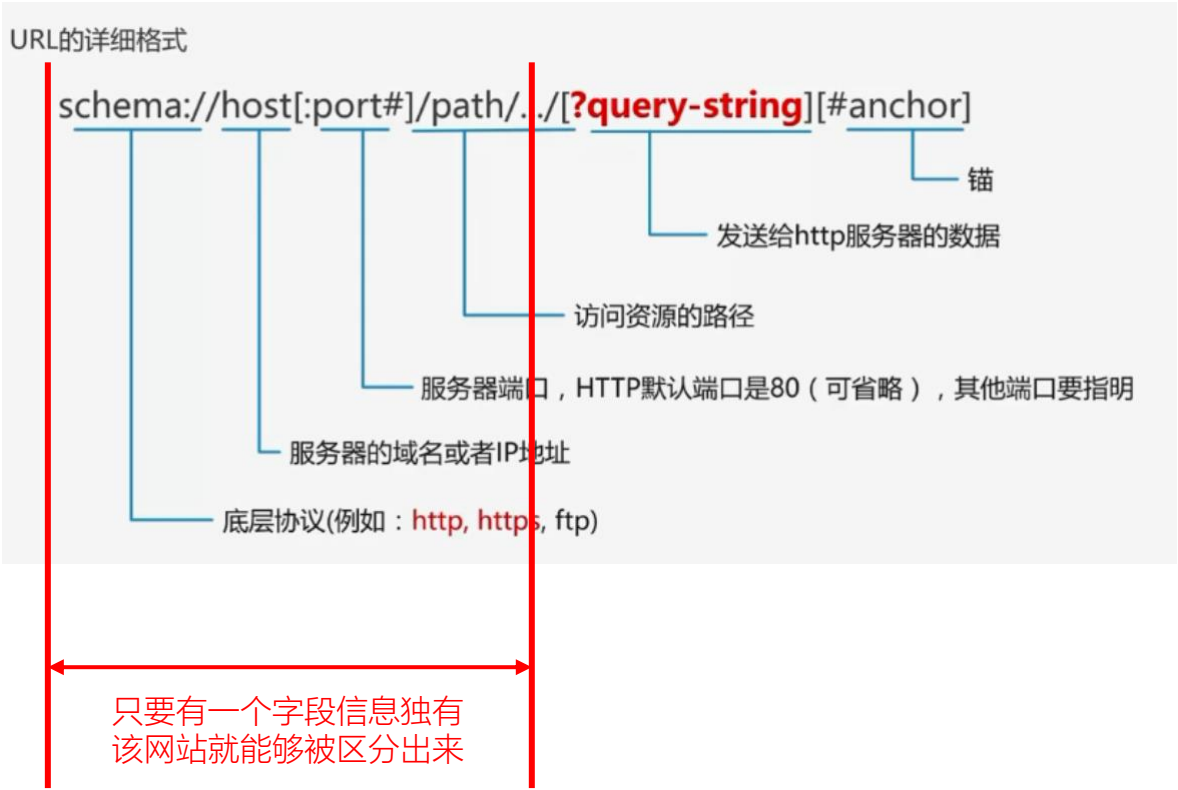
任务目标：

- 安装并配置Apache HTTP Server。
- 通过默认网站、虚拟目录、端口号、域名四种方式发布静态网站。

操作步骤：

- 安装并配置Apache HTTP Server。
- 创建静态网站。
- 配置Apache HTTP Server进行网站发布。
  - 发布默认网站。
  - 发布多个静态网站。
- 在本地主机进行测试。

操作演示：


URL的详细格式
schema://host[:port#]/path/.../[?query-string][#anchor]
锚
发送给http服务器的数据
访问资源的路径
服务器端口，HTTP默认端口是80（可省略），其他端口要指明
服务器的域名或者IP地址
底层协议(例如：http, https, ftp)
只要有一个字段信息独有
该网站就能够被区分出来

2. 发布静态网站

- 在openEuler中，Apache配置文件的存放位置是/etc/httpd目录。
  - /etc/httpd/conf/httpd.conf
    - 是主配置文件，Apache的配置主要是使用该文件。
  - /etc/httpd/conf.d
    - 是额外配置文件目录，如果不想修改主配置文件httpd.conf，可在此将配置独立出来。Apache启动时会将该目录下配置信息与主配置文件信息合并后执行。
  - /etc/httpd/modules
    - 用于存放所有已安装的模块。
  - /etc/httpd/logs
    - 用于存放日志文件，此文件是链接文件。
  - /var/www/html
    - 是默认网站存放的目录。

2. 发布静态网站

- Apache HTTP Server日志文件记录了Apache HTTP Server的运行历史，通过管理和分析日志可及时了解 Apache HTTP Server的运行状态。
- Apache包含访问日志和错误日志两个部分。
  - 日志文件在 openEuler 中的存放位置是/etc/httpd/logs目录。
  - 访问日志的文件名为access.log。
  - 错误日志的文件名为error.log。
- 如果使用SSL服务，日志文件将包括关于SSL运行的日志文件。
  - 日志文件分别是ssl_access_log、ssl_error_log和ssl_request_log 共3个文件。

2. 发布静态网站

- Apache日志格式
  可通过httpd.conf文件进行设置。

表 4-2-2　日志格式配置中常用字符串含义

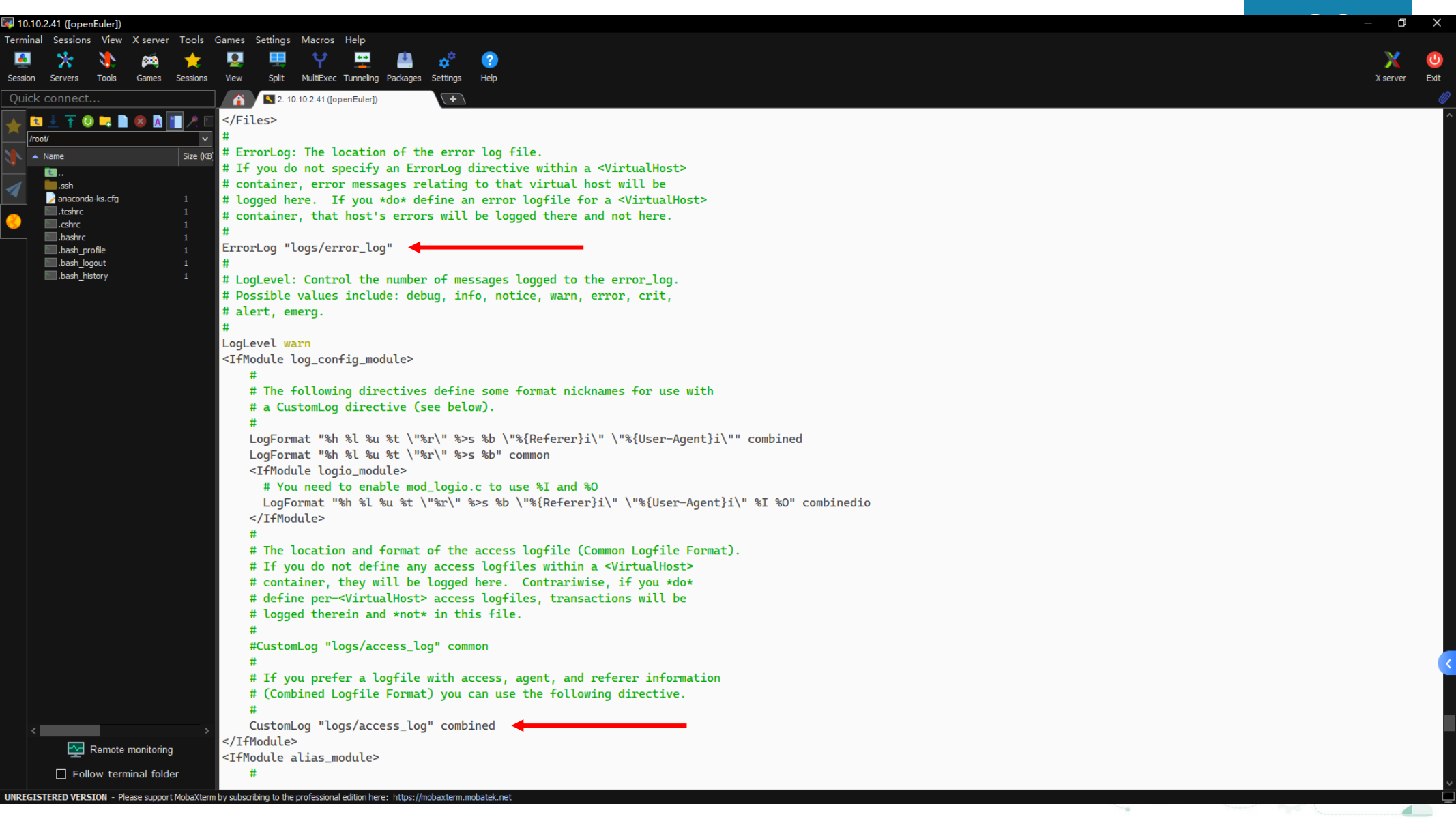
变量	含义
%%	百分号
%a	请求客户端的 IP 地址
%A	本机 IP 地址
%B	不包含 HTTP 头的已发送字节数
%b	不包含 HTTP 头的 CLF 格式的已发送字节数量。当没有发送数据时，显示“-”而不是 0
%D	服务器处理本请求所用时间，单位为μs
%f	文件名
%h	远端主机
%H	请求使用的协议
%l	远程登录名
%m	请求的方法
%{VARNAME}C	在请求中传送给服务端的 cookie VARNAME 的内容
%{VARNAME}e	环境变量 VARNAME 的值
%{VARNAME}i	发送到服务器的请求头 VARNAME 的内容
%{VARNAME}n	其他模块注释 VARNAME 的内容
%{VARNAME}o	应答头 VARNAME 的内容
%p	服务器响应请求时使用的端口
%P	响应请求的子进程 ID
%q	查询字符串（如果存在查询字符串，则包含“?”后面的部分；否则，它是一个空字符串）
%r	请求的第一行
%s	状态。对于内部重定向的请求，这里指原来请求的状态；如果用%...>s，则是指后来的请求
%t	接收请求的时间，如：18/Sep/2019:19:18:28 -0400
%{format}t	以指定格式 format 表示的时间
%T	为响应请求而耗费的时间，单位为 s
%u	远程用户
%U	用户所请求的 URL 路径
%v	响应请求的服务器的 ServerName
%V	依照 UseCanonicalName 设置得到的服务器名字
%I	接收的字节数，包含头与正文
%O	发送的字节数，包含头与正文

2. 发布静态网站

- Apache使用LogLevel定义记录错误日志的等级标准。
  - 不同级别日志记录详细程度不同，比如说当指定级别为error时，crit、alert、emerg信息也会被记录。

表 4-2-3　错误日志记录等级

等级	说明
emerg	紧急，系统无法使用
alert	必须立即采取措施
crit	关键错误，危险情况的警告，由于配置不当所致
error	一般错误
warn	警告信息，不算是错误信息，主要记录服务器出现的某种信息
notice	需要引起注意的情况
info	值得报告的一般消息，比如服务器重启
debug	由运行 debug 模式的程序所产生的消息


10.10.2.41 ([openEuler])
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 10.10.2.41 ([openEuler])
/root/
Name Size (KB)
..
.ssh
anaconda-ks.cfg 1
.tcshrc 1
.cshrc 1
.bashrc 1
.bash_profile 1
.bash_logout 1
.bash_history 1
Remote monitoring
Follow terminal folder
</Files>
#
ErrorLog: The location of the error log file.
If you do not specify an ErrorLog directive within a <VirtualHost>
container, error messages relating to that virtual host will be
logged here. If you *do* define an error logfile for a <VirtualHost>
container, that host's errors will be logged there and not here.
#
ErrorLog "logs/error_log"
#
LogLevel: Control the number of messages logged to the error_log.
Possible values include: debug, info, notice, warn, error, crit,
alert, emerg.
#
LogLevel warn
<IfModule log_config_module>
#
The following directives define some format nicknames for use with
a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
<IfModule logio_module>
You need to enable mod_logio.c to use %I and %O
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio
</IfModule>
#
The location and format of the access logfile (Common Logfile Format).
If you do not define any access logfiles within a <VirtualHost>
container, they will be logged here. Contrariwise, if you *do*
define per-<VirtualHost> access logfiles, transactions will be
logged therein and *not* in this file.
#
#CustomLog "logs/access_log" common
#
If you prefer a logfile with access, agent, and referer information
(Combined Logfile Format) you can use the following directive.
#
CustomLog "logs/access_log" combined
</IfModule>
<IfModule alias_module>
#
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

3. 建设内容网站服务

使用WordPress建设内容网站服务

任务目标：

- 实现LAMP。
- 使用WordPress建设内容网站服务。

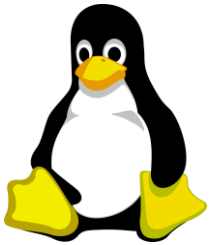
操作步骤：

- 部署LAMP。
- 获取WordPress。
- 部署WordPress，完成初始化安全。
- 发布内容并进行测试。

操作演示：

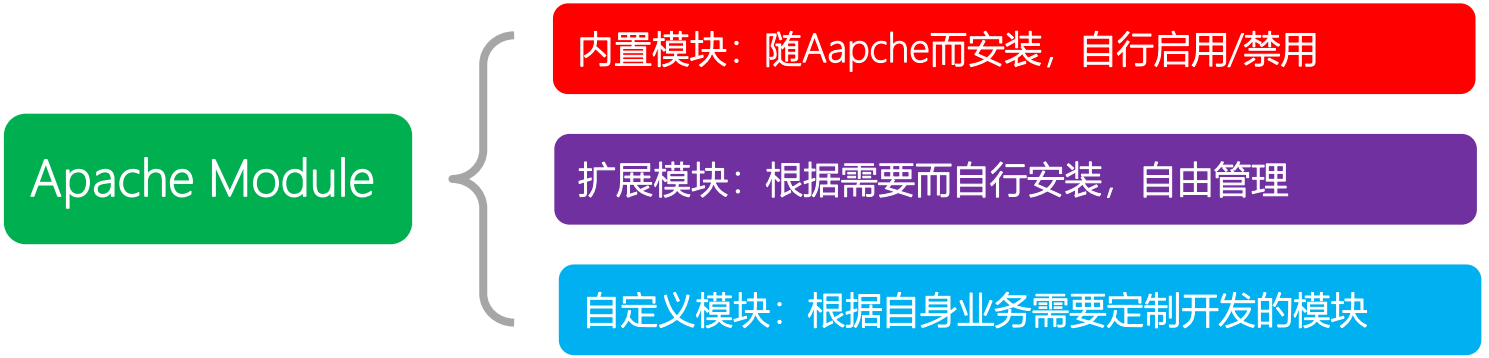
3. 建设内容网站服务

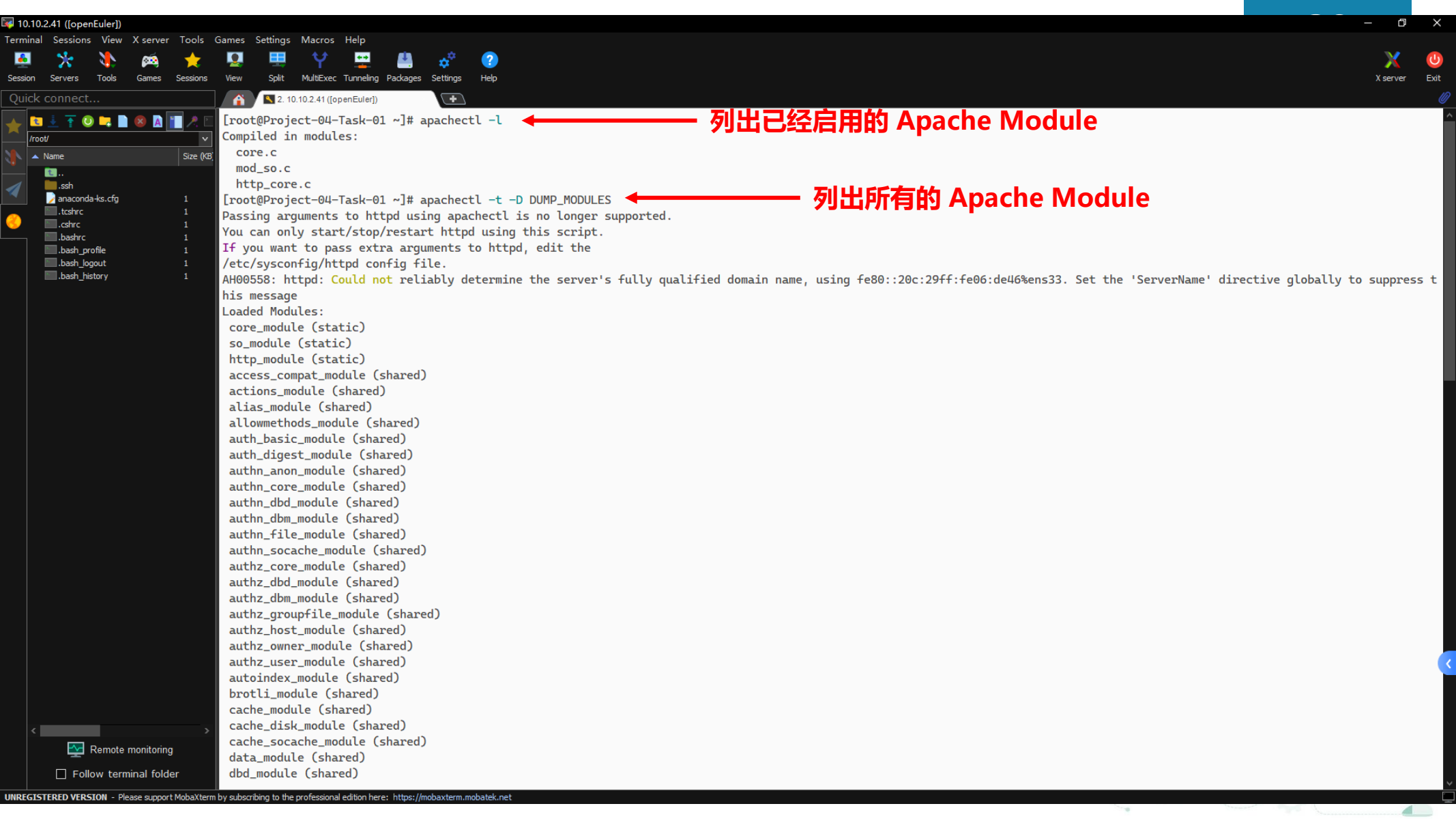
- LAMP是发布PHP程序的开源稳定架构，由Linux作为操作系统、Apache作为网站服务器、MySQL/MariaDB作为数据库管理系统、PHP/Perl/Python作为服务器端脚本解释器。
- 本项目使用openEuler操作系统，使用Apache网站服务器，使用MariaDB数据库管理系统，使用PHP服务器端脚本解释器，实现LAMP架构的网站服务器部署。

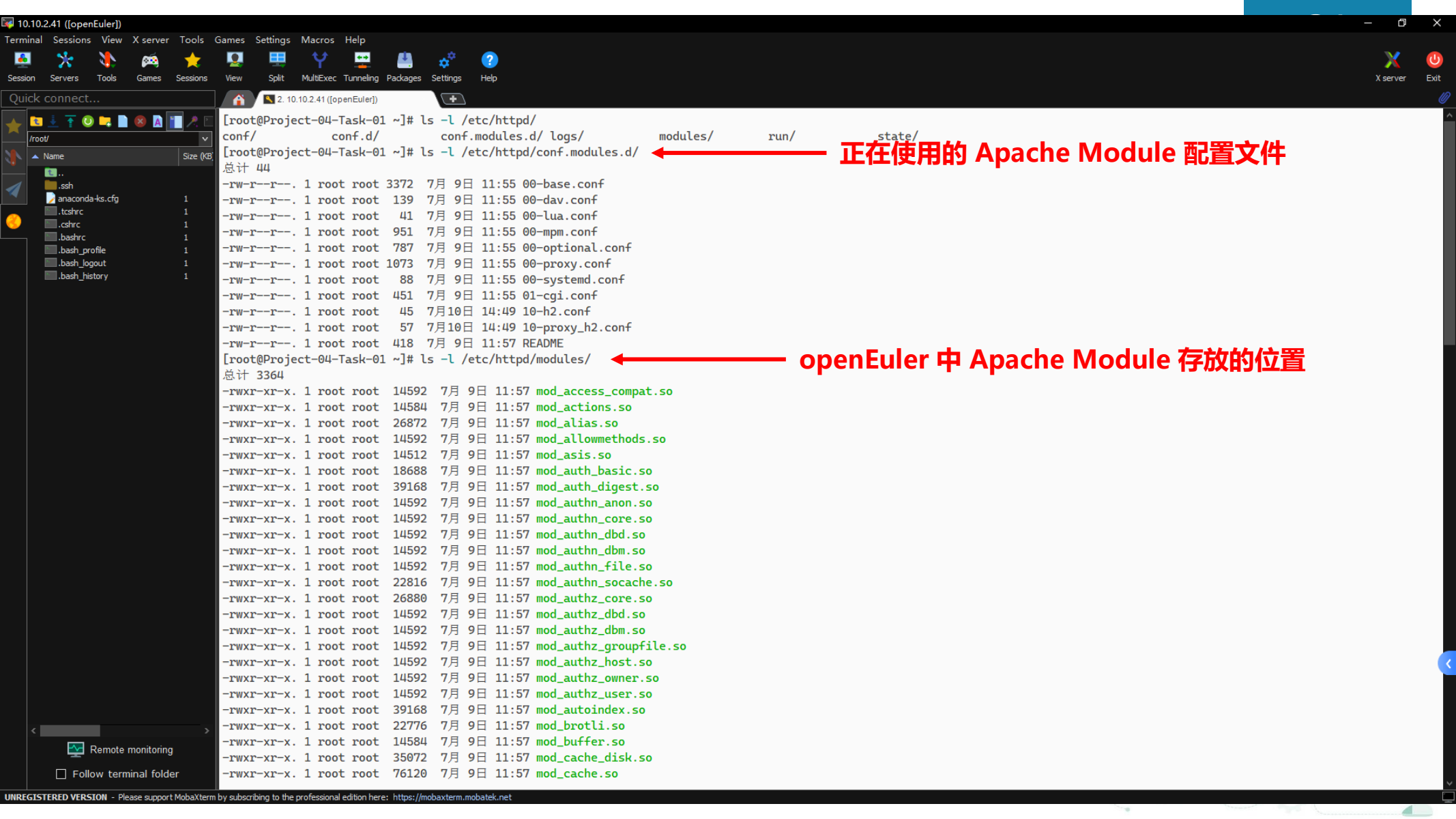


4. 查看Apache的服务状态

- Apache是模块化的设计，大多数功能被分散到各模块中，各模块在系统启动时按需载入。
  - 安装 Apache 时会默认安装一些模块，如果需要实现某种特定的功能可以根据实际需求自行安装 Apache 模块。
  - Apache 模块只与 Apache 版本有关，和操作系统无关。




10.10.2.41 ([openEuler])
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 10.10.2.41 ([openEuler])
/root/
Name Size (KB)
..
.ssh
anaconda-ks.cfg 1
.tcshrc 1
.cshrc 1
.bashrc 1
.bash_profile 1
.bash_logout 1
.bash_history 1
Remote monitoring
Follow terminal folder
[root@Project-04-Task-01 ~]# apachectl -l
Compiled in modules:
core.c
mod_so.c
http_core.c
[root@Project-04-Task-01 ~]# apachectl -t -D DUMP_MODULES
Passing arguments to httpd using apachectl is no longer supported.
You can only start/stop/restart httpd using this script.
If you want to pass extra arguments to httpd, edit the
/etc/sysconfig/httpd config file.
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::20c:29ff:fe06:de46%ens33. Set the 'ServerName' directive globally to suppress t
his message
Loaded Modules:
core_module (static)
so_module (static)
http_module (static)
access_compat_module (shared)
actions_module (shared)
alias_module (shared)
allowmethods_module (shared)
auth_basic_module (shared)
auth_digest_module (shared)
authn_anon_module (shared)
authn_core_module (shared)
authn_dbd_module (shared)
authn_dbm_module (shared)
authn_file_module (shared)
authn_socache_module (shared)
authz_core_module (shared)
authz_dbd_module (shared)
authz_dbm_module (shared)
authz_groupfile_module (shared)
authz_host_module (shared)
authz_owner_module (shared)
authz_user_module (shared)
autoindex_module (shared)
brotli_module (shared)
cache_module (shared)
cache_disk_module (shared)
cache_socache_module (shared)
data_module (shared)
dbd_module (shared)
列出已经启用的 Apache Module
列出所有的 Apache Module
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net


10.10.2.41 ([openEuler])
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 10.10.2.41 ([openEuler])
/root/
Name Size (KB)
..
.ssh
anaconda-ks.cfg 1
.tcshrc 1
.cshrc 1
.bashrc 1
.bash_profile 1
.bash_logout 1
.bash_history 1
Remote monitoring
Follow terminal folder
[root@Project-04-Task-01 ~]# ls -l /etc/httpd/
conf/ conf.d/ conf.modules.d/ logs/ modules/ run/ state/
[root@Project-04-Task-01 ~]# ls -l /etc/httpd/conf.modules.d/
正在使用的 Apache Module 配置文件
总计 44
-rw-r--r--. 1 root root 3372 7月 9日 11:55 00-base.conf
-rw-r--r--. 1 root root 139 7月 9日 11:55 00-dav.conf
-rw-r--r--. 1 root root 41 7月 9日 11:55 00-lua.conf
-rw-r--r--. 1 root root 951 7月 9日 11:55 00-mpm.conf
-rw-r--r--. 1 root root 787 7月 9日 11:55 00-optional.conf
-rw-r--r--. 1 root root 1073 7月 9日 11:55 00-proxy.conf
-rw-r--r--. 1 root root 88 7月 9日 11:55 00-systemd.conf
-rw-r--r--. 1 root root 451 7月 9日 11:55 01-cgi.conf
-rw-r--r--. 1 root root 45 7月10日 14:49 10-h2.conf
-rw-r--r--. 1 root root 57 7月10日 14:49 10-proxy_h2.conf
-rw-r--r--. 1 root root 418 7月 9日 11:57 README
[root@Project-04-Task-01 ~]# ls -l /etc/httpd/modules/
openEuler 中 Apache Module 存放的位置
总计 3364
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_access_compat.so
-rwxr-xr-x. 1 root root 14584 7月 9日 11:57 mod_actions.so
-rwxr-xr-x. 1 root root 26872 7月 9日 11:57 mod_alias.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_allowmethods.so
-rwxr-xr-x. 1 root root 14512 7月 9日 11:57 mod_asis.so
-rwxr-xr-x. 1 root root 18688 7月 9日 11:57 mod_auth_basic.so
-rwxr-xr-x. 1 root root 39168 7月 9日 11:57 mod_auth_digest.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authn_anon.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authn_core.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authn_dbd.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authn_dbm.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authn_file.so
-rwxr-xr-x. 1 root root 22816 7月 9日 11:57 mod_authn_socache.so
-rwxr-xr-x. 1 root root 26880 7月 9日 11:57 mod_authz_core.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_dbd.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_dbm.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_groupfile.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_host.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_owner.so
-rwxr-xr-x. 1 root root 14592 7月 9日 11:57 mod_authz_user.so
-rwxr-xr-x. 1 root root 39168 7月 9日 11:57 mod_autoindex.so
-rwxr-xr-x. 1 root root 22776 7月 9日 11:57 mod_brotli.so
-rwxr-xr-x. 1 root root 14584 7月 9日 11:57 mod_buffer.so
-rwxr-xr-x. 1 root root 35072 7月 9日 11:57 mod_cache_disk.so
-rwxr-xr-x. 1 root root 76120 7月 9日 11:57 mod_cache.so
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

4. 查看Apache的服务状态

- Apache Module mod_status主要用于提供服务器运行状态的信息。
  - 服务器状态监控
    - 实时显示服务器的当前连接数、正在处理的请求数、已处理的请求数等信息，实现快速了解服务器的负载情况。
    - 可以显示每个连接的详细信息，包括客户端 IP 地址、请求方法、请求 URL 等，理解业务的服务情况。
  - 性能分析
    - 实时显示不同时间段的连接数和请求处理情况，以分析服务器的性能瓶颈。
    - 可以查看服务器的响应时间，了解服务器对请求的处理速度。

Apache Module mod_status

Available Languages: en | fr | ja | ko | tr

Description:	Provides information on server activity and performance
Status:	Base
Module Identifier:	status_module
Source File:	mod_status.c

Summary

The Status module allows a server administrator to find out how well their server is performing. A HTML page is presented that gives the current server statistics in an easily readable form. If required this page can be made to automatically refresh (given a compatible browser). Another page gives a simple machine-readable list of the current server state.

The details given are:

- The number of workers serving requests
- The number of idle workers
- The status of each worker, the number of requests that worker has performed and the total number of bytes served by the worker (*)
- A total number of accesses and byte count served (*)
- The time the server was started/restarted and the time it has been running for
- Averages giving the number of requests per second, the number of bytes served per second and the average number of bytes per request (*)
- The current percentage CPU used by each worker and in total by all workers combined (*)
- The current hosts and requests being processed (*)

The lines marked "(*)" are only available if `ExtendedStatus` is `On`. In version 2.3.6, loading mod_status will toggle `ExtendedStatus` On by default.

Enabling Status Support

To enable status reports only for browsers from the example.com domain add this code to your `httpd.conf` configuration file

```
<Location "/server-status">
    SetHandler server-status
    Require host example.com
</Location>
```

You can now access server statistics by using a Web browser to access the page `http://your.server.name/server-status`

Automatic Updates

You can get the status page to update itself automatically if you have a browser that supports "refresh". Access the page `http://your.server.name/server-status?refresh=N` to refresh the page every N seconds.

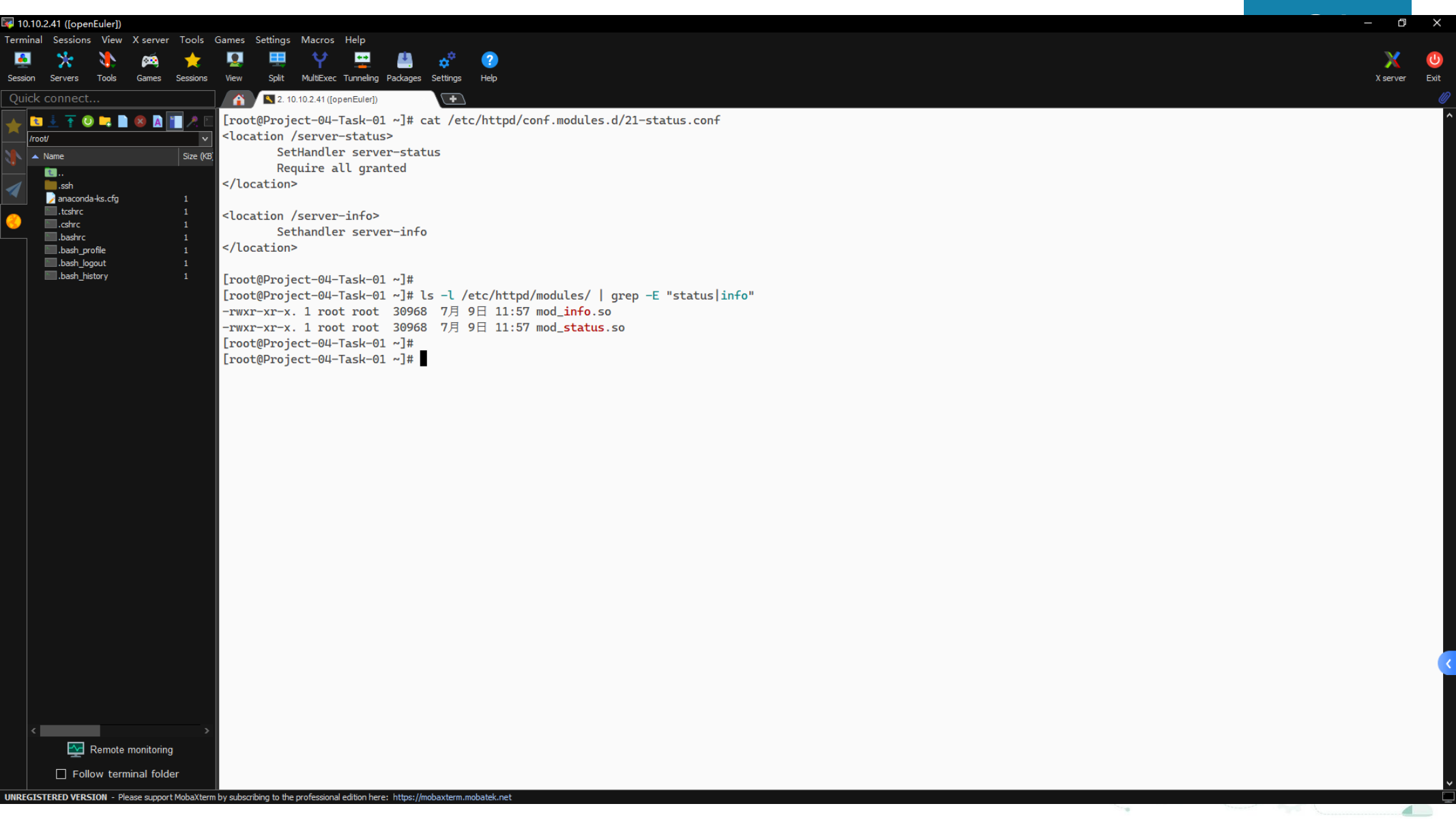
Machine Readable Status File

A machine-readable version of the status file is available by accessing the page `http://your.server.name/server-status?auto`. This is useful when automatically run, see the Perl program `log_server_status`, which you will find in the `/support` directory of your Apache HTTP Server installation.

It should be noted that if `mod_status` is loaded into the server, its handler capability is available in *all* configuration files, including *per*-directory files (*e.g.*, `.htaccess`). This may have security-related ramifications for your site.

Using server-status to troubleshoot

The `server-status` page may be used as a starting place for troubleshooting a situation where your server is consuming all available resources (CPU or memory), and you wish to identify which requests or clients are causing the problem.

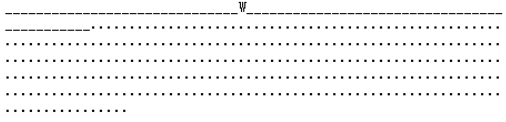

10.10.2.41 ([openEuler])
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 10.10.2.41 ([openEuler])
/root/
Name Size (KB)
..
.ssh
anaconda-ks.cfg 1
.tcshrc 1
.cshrc 1
.bashrc 1
.bash_profile 1
.bash_logout 1
.bash_history 1
[root@Project-04-Task-01 ~]# cat /etc/httpd/conf.modules.d/21-status.conf
<location /server-status>
SetHandler server-status
Require all granted
</location>
<location /server-info>
Sethandler server-info
</location>
[root@Project-04-Task-01 ~]#
[root@Project-04-Task-01 ~]# ls -l /etc/httpd/modules/ | grep -E "status|info"
-rwxr-xr-x. 1 root root 30968 7月 9日 11:57 mod_info.so
-rwxr-xr-x. 1 root root 30968 7月 9日 11:57 mod_status.so
[root@Project-04-Task-01 ~]#
[root@Project-04-Task-01 ~]#
Remote monitoring
Follow terminal folder
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:45:11 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 2 minutes 42 seconds
Server load: 0.03 0.02 0.00
Total accesses: 2 - Total Traffic: 5 kB - Total Duration: 1
CPU Usage: u.04 s.05 cu0 cs0 - .0556% CPU load
.0123 requests/sec - 31 B/second - 2560 B/request - .5 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 74 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	0	0	25	0	0	0
1	2006	no	0	yes	1	0	24	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
Sum	3	0	0		1	0	74	0	0	0



Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	0/1/1	_	0.02	12	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	12	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
1-0	2006	1/0/0	W	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Srv Child Server number - generation
PID OS process ID

- The Status module allows a server administrator to find out how well their server is performing. A HTML page is presented that gives the current server statistics in an easily readable form. If required this page can be made to automatically refresh (given a compatible browser). Another page gives a simple machine-readable list of the current server state.
- The details given are:
  - The number of workers serving requests
  - The number of idle workers
  - The status of each worker, the number of requests that worker has performed and the total number of bytes served by the worker (*)
  - A total number of accesses and byte count served (*)
  - The time the server was started/restarted and the time it has been running for
  - Averages giving the number of requests per second, the number of bytes served per second and the average number of bytes per request (*)
  - The current percentage CPU used by each worker and in total by all workers combined (*)
  - The current hosts and requests being processed (*)

Apache Status

不安全 | 10.10.2.41/server-status

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:45:11 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 2 minutes 42 seconds
Server load: 0.03 0.02 0.00
Total accesses: 2 - Total Traffic: 5 kB - Total Duration: 1
CPU Usage: u.04 s.05 cu0 cs0 - .0556% CPU load
.0123 requests/sec - 31 B/second - 2560 B/request - .5 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 74 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	0	0	25	0	0	0
1	2006	no	0	yes	1	0	24	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
Sum	3	0	0		1	0	74	0	0	0

```
______________________________W_________________________________
___________.....................................................
................................................................
................................................................
................................................................
................................................................
...............
```

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	0/1/1	_	0.02	12	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	12	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
1-0	2006	1/0/0	**W**	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Srv Child Server number - generation
PID OS process ID

Apache Server Status三种访问方式：

- 常规方式：http://your.server.name/server-status
- 自动刷新：http://your.server.name/server-status?refresh=N
- 机器阅读： http://your.server.name/server-status?auto

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:45:11 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 2 minutes 42 seconds
Server load: 0.03 0.02 0.00
Total accesses: 2 - Total Traffic: 5 kB - Total Duration: 1
CPU Usage: u.04 s.05 cu0 cs0 - .0556% CPU load
.0123 requests/sec - 31 B/second - 2560 B/request - .5 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 74 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	0	0	25	0	0	0
1	2006	no	0	yes	1	0	24	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
Sum	3	0	0		1	0	74	0	0	0

_____W_____
_____.....
.....
.....
.....
.....
.....
.....

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	0/1/1	_	0.02	12	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	12	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
1-0	2006	1/0/0	W	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Srv Child Server number - generation
PID OS process ID

- Server Version — Apache Http Server 软件版本
- Server MPM — Apache 的工作模式
- Server Built — Apache 当前版本的编译时间

- Current Time — 当前系统时间
- Restart Time — 最近重启httpd服务的时间
- Parent Server Config. Generation — httpd的父进程重启的次数
- Parent Server MPM Generation — httpd的event模式中主进程重启的次数

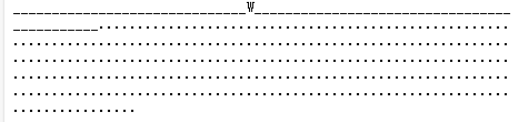
- Server uptime — 最近启动httpd到现在共计运行的时长
- Server load — 服务器负载，1min 5min 15min 时间范围内

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:45:11 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 2 minutes 42 seconds
Server load: 0.03 0.02 0.00
Total accesses: 2 - Total Traffic: 5 kB - Total Duration: 1
CPU Usage: u.04 s.05 cu0 cs0 - .0556% CPU load
.0123 requests/sec - 31 B/second - 2560 B/request - .5 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 74 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	0	0	25	0	0	0
1	2006	no	0	yes	1	0	24	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
Sum	3	0	0		1	0	74	0	0	0



Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	0/1/1	_	0.02	12	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	12	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
1-0	2006	1/0/0	W	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Srv Child Server number - generation
PID OS process ID

- Total accesses　　httpd启动后接收的请求总数
- Total Traffic　　httpd启动后传送的数据总量
- Total Duration　　httpd启动后使用CPU的总时长

- CPU Usage　　CPU使用情况

- .015 requests/sec - 91 B/second - 5.9 kB/request - .35 ms/request
  httpd启动后的平均负载
  每秒多少个请求
  每秒传送多少数据
  每个请求平均传送多少数据
  每个请求平均耗时多长

- 1 requests currently being processed, 74 idle workers
  当前占用的线程数
  当前空闲的线程数

Apache Status

10.10.2.41/server-status

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:55:04 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 12 minutes 35 seconds
Server load: 0.00 0.00 0.00
Total accesses: 4 - Total Traffic: 141 kB - Total Duration: 7
CPU Usage: u.07 s.08 cu0 cs0 - .0199% CPU load
.0053 requests/sec - 191 B/second - 35.3 kB/request - 1.75 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 99 idle workers

```
[root@Project-04-Task-01 ~]# ps aux | grep httpd
root        2002  0.0  2.6  18896 11364 ?        Ss   17:42   0:00 /usr/sbin/httpd -DFOREGROUND
apache      2004  0.0  1.6  18924  6964 ?        S    17:42   0:00 /usr/sbin/httpd -DFOREGROUND
apache      2005  0.0  2.3 1109616 10188 ?       Sl   17:42   0:00 /usr/sbin/httpd -DFOREGROUND
apache      2006  0.0  2.3 978480  9848 ?        Sl   17:42   0:00 /usr/sbin/httpd -DFOREGROUND
apache      2007  0.0  1.9 978480  8540 ?        Sl   17:42   0:00 /usr/sbin/httpd -DFOREGROUND
apache      2208  0.0  1.9 978480  8540 ?        Sl   17:46   0:00 /usr/sbin/httpd -DFOREGROUND
root        2284  0.0  0.5  21984  2404 pts/0    S+   17:53   0:00 grep --color=auto httpd
[root@Project-04-Task-01 ~]#
```

Slot	PID	Stopping	Connections		Threads		Async connections		
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	1	0	24	0	0	0
1	2006	no	0	yes	0	0	25	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
3	2208	no	0	yes	0	0	25	0	0	0
Sum	4	0	0		1	0	99	0	0	0

```
W_______________________________________________________________
____________________________________............................
................................................................
................................................................
................................................................
................................................................
................
```

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	1/0/0	**W**	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	605	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	605	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
0-0	2005	0/1/1	_	0.03	570	4	4	0.0	0.13	0.13	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-info HTTP/1.1
1-0	2006	0/1/1		0.02	592	0	0	0.0	0.01	0.01	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Apache Status

不安全 | 10.10.2.41/server-status

Apache Server Status for 10.10.2.41 (via 10.10.2.41)

Server Version: Apache/2.4.58 (Unix)
Server MPM: event
Server Built: Jul 9 2024 11:57:06

Current Time: Sunday, 13-Oct-2024 17:55:04 CST
Restart Time: Sunday, 13-Oct-2024 17:42:29 CST
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 12 minutes 35 seconds
Server load: 0.00 0.00 0.00
Total accesses: 4 - Total Traffic: 141 kB - Total Duration: 7
CPU Usage: u.07 s.08 cu0 cs0 - .0199% CPU load
.0053 requests/sec - 191 B/second - 35.3 kB/request - 1.75 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 99 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	1	0	24	0	0	0
1	2006	no	0	yes	0	0	25	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
3	2208	no	0	yes	0	0	25	0	0	0
Sum	4	0	0		1	0	99	0	0	0

W_____
_____.....
.....
.....
.....
.....
.....
.....

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	1/0/0	W	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	605	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	605	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
0-0	2005	0/1/1	_	0.03	570	4	4	0.0	0.13	0.13	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-info HTTP/1.1
1-0	2006	0/1/1		0.02	592	0	0	0.0	0.01	0.01	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

- Scoreboard Key:
  - "_" Waiting for Connection 等待连接中
  - "S" Starting up 启动中
  - "R" Reading Request 正在读取请求
  - "W" Sending Reply 正在发送响应
  - "K" Keepalive (read) 保持联机的状态
  - "D" DNS Lookup 正在查找DNS
  - "C" Closing connection 正在关闭连接
  - "L" Logging 正在写入日志
  - "G" Gracefully finishing 进入正常结束程序
  - "I" Idle cleanup of worker 处理限制
  - "." Open slot with no current process 没有创建该线程

Apache Status

不安全 | 10.10.2.41/server-status

搜索

.00585 requests/sec - 176 B/second - 29.4 kB/request - 1.6 ms/request
1 requests currently being processed, 0 workers gracefully restarting, 99 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections			
			total	accepting	busy	graceful	idle	writing	keep-alive	closing
0	2005	no	0	yes	1	0	24	0	0	0
1	2006	no	0	yes	0	0	25	0	0	0
2	2007	no	0	yes	0	0	25	0	0	0
3	2208	no	0	yes	0	0	25	0	0	0
Sum	4	0	0		1	0	99	0	0	0

```
__W_____________________________________________________________
____________________________________............................
................................................................
................................................................
................................................................
................
```

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2005	0/1/1	_	0.11	99	1	1	0.0	0.01	0.01	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	1/0/0	**W**	0.00	0	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	704	1	1	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1
0-0	2005	0/1/1	_	0.02	704	0	0	0.0	0.00	0.00	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /favicon.ico HTTP/1.1
0-0	2005	0/1/1	_	0.03	669	4	4	0.0	0.13	0.13	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-info HTTP/1.1
1-0	2006	0/1/1	_	0.02	692	0	0	0.0	0.01	0.01	10.10.2.1	http/1.1	fe80::20c:29ff:fe06:de46%ens33:	GET /server-status HTTP/1.1

Srv	Child Server number - generation
PID	OS process ID
Acc	Number of accesses this connection / this child / this slot
M	Mode of operation
CPU	CPU usage, number of seconds
SS	Seconds since beginning of most recent request
Req	Milliseconds required to process most recent request
Dur	Sum of milliseconds required to process all requests
Conn	Kilobytes transferred this connection
Child	Megabytes transferred this child
Slot	Total megabytes transferred this slot

- A connection is a link between a client and a child which resides in a slot.
- 连接是客户端与位于插槽中的子进程（或子线程）之间的链接。
- A child is a process or a thread that handles requests.
- 子进程（或子线程）是处理请求的进程或线程。
- A slot is a placeholder for a child. There may or may not be a child there depending on max/min spare servers.
- 插槽是子进程（或子线程）的占位符。根据最大 / 最小空闲服务器数量的不同，可能有子进程（或子线程），也可能没有。

- Srv　Child Server number - generation
- PID　OS process ID 处理该请求的进程的PID
- Acc　Number of accesses this connection / this child / this slot
  当前连接 / 当前连接所对应的子进程 /
  当前连接所对应的子进程所属槽的读取次数
- M　Mode of operation 目前的状态
- CPU　CPU usage, number of seconds 使用的CPU资源
- SS　Seconds since beginning of most recent request
  距离上次处理要求的时长（多少秒）
- Req　Milliseconds required to process most recent request
  最后处理操作所耗费的时间（多少毫秒）
- Dur　Sum of milliseconds required to process all requests
- Conn　Kilobytes transferred this connection
  当前连接传送的数据量
- Child　Megabytes transferred this child
  当前连接所对应的子进程传送的数据量
- Slot　Total megabytes transferred this slot
  当前连接所对应的子进程所属槽传送的数据量
- Client　客户端IP地址
- Protocol　使用的协议
- VHost　Apache服务器的IP地址和接收请求的端口
- Request　请求的路径，使用的协议

4. 查看Apache的服务状态

- Apache Module mod_info提供了一个服务器信息页面，通过访问特定的URL可以获取关于服务器配置、已加载模块等详细信息。
- 通常可以显示诸如服务器版本、编译选项、加载的模块列表以及一些特定的配置参数等信息。

Apache Module mod_info

Available Languages: en | fr | ja | ko

Description:	Provides a comprehensive overview of the server configuration
Status:	Extension
Module Identifier:	info_module
Source File:	mod_info.c

Summary

To configure mod_info, add the following to your httpd.conf file.

```
<Location "/server-info">
    SetHandler server-info
</Location>
```

You may wish to use mod_authz_host inside the <Location> directive to limit access to your server configuration information:

```
<Location "/server-info">
    SetHandler server-info
    Require host example.com
</Location>
```

Once configured, the server information is obtained by accessing http://your.host.example.com/server-info

Security Issues

Once mod_info is loaded into the server, its handler capability is available in *all* configuration files, including per-directory files (*e.g.*, .htaccess). This may have security-related ramifications for your site.

In particular, this module can leak sensitive information from the configuration directives of other Apache modules such as system paths, usernames/passwords, database names, etc. Therefore, this module should **only** be used in a controlled environment and always with caution.

You will probably want to use mod_authz_host to limit access to your server configuration information.

Access control

```
<Location "/server-info">
    SetHandler server-info
    # Allow access from server itself
    Require ip 127.0.0.1

    # Additionally, allow access from local workstation
    Require ip 192.168.1.17
</Location>
```

Selecting the information shown

By default, the server information includes a list of all enabled modules, and for each module, a description of the directives understood by that module, the hooks implemented by that module, and the relevant directives from the current configuration.

Other views of the configuration information are available by appending a query to the server-info request. For example, http://your.host.example.com/server-info?config will show all configuration directives.

?<module-name>
: Only information relevant to the named module

Topics
- Security Issues
- Selecting the information shown
- Dumping the configuration on startup
- Known Limitations

Directives
AddModuleInfo

Bugfix checklist
- httpd changelog
- Known issues
- Report a bug

See also
- Comments

Apache Server Information

Subpages:
Configuration Files, Server Settings, Module List, Active Hooks, Available Providers

Sections:
Loaded Modules, Server Settings, Startup Hooks, Request Hooks, Other Hooks, Providers

导航

Loaded Modules

模块导航

core.c, event.c, http_core.c, mod_access_compat.c, mod_actions.c, mod_alias.c, mod_allowmethods.c, mod_auth_basic.c, mod_auth_digest.c, mod_authn_anon.c, mod_authn_core.c, mod_authn_dbd.c, mod_authn_dbm.c, mod_authn_file.c, mod_authn_socache.c, mod_authz_core.c, mod_authz_dbd.c, mod_authz_dbm.c, mod_authz_groupfile.c, mod_authz_host.c, mod_authz_owner.c, mod_authz_user.c, mod_autoindex.c, mod_brotli.c, mod_cache.c, mod_cache_disk.c, mod_cache_socache.c, mod_cgid.c, mod_data.c, mod_dav.c, mod_dav_fs.c, mod_dav_lock.c, mod_dbd.c, mod_deflate.c, mod_dir.c, mod_dumpio.c, mod_echo.c, mod_env.c, mod_expires.c, mod_ext_filter.c, mod_filter.c, mod_headers.c, mod_http2.c, mod_include.c, mod_info.c, mod_lbmethod_bybusyness.c, mod_lbmethod_byrequests.c, mod_lbmethod_bytraffic.c, mod_lbmethod_heartbeat.c, mod_log_config.c, mod_logio.c, mod_lua.c, mod_macro.c, mod_mime.c, mod_mime_magic.c, mod_negotiation.c, mod_proxy.c, mod_proxy_ajp.c, mod_proxy_balancer.c, mod_proxy_connect.c, mod_proxy_express.c, mod_proxy_fcgi.c, mod_proxy_fdpass.c, mod_proxy_ftp.c, mod_proxy_hcheck.c, mod_proxy_http.c, mod_proxy_http2.c, mod_proxy_scgi.c, mod_proxy_uwsgi.c, mod_proxy_wstunnel.c, mod_remoteip.c, mod_reqtimeout.c, mod_request.c, mod_rewrite.c, mod_setenvif.c, mod_slotmem_plain.c, mod_slotmem_shm.c, mod_so.c, mod_socache_dbm.c, mod_socache_memcache.c, mod_socache_redis.c, mod_socache_shmcb.c, mod_status.c, mod_substitute.c, mod_suexec.c, mod_systemd.c, mod_unique_id.c, mod_unixd.c, mod_userdir.c, mod_version.c, mod_vhost_alias.c, mod_watchdog.c,

Server Settings

Server Version: Apache/2.4.58 (Unix)
Server Built: Jul 9 2024 11:57:06
Server loaded APR Version: 1.7.4
Compiled with APR Version: 1.7.4
Server loaded APU Version: 1.6.3
Compiled with APU Version: 1.6.3
Server loaded PCRE Version: 10.42 2022-12-11
Compiled with PCRE Version: 10.42 2022-12-11
Module Magic Number: 20120211:134
Hostname/port: 10.10.2.41:80
Timeouts: connection: 60 keep-alive: 5
MPM Name: event
MPM Information: Max Daemons: 1 Threaded: yes Forked: yes
Server Architecture: 64-bit
Server Root: /etc/httpd
Config File: /etc/httpd/conf/httpd.conf
Server Built With:
-D APR_HAS_SENDFILE
-D APR_HAS_MMAP
-D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
-D APR_USE_PROC_PTHREAD_SERIALIZE
-D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
-D APR_HAS_OTHER_CHILD
-D AP_HAVE_RELIABLE_PIPED_LOGS
-D HTTPD_ROOT="/etc/httpd"
-D SUEXEC_BIN="/usr/sbin/suexec"
-D DEFAULT_PIDLOG="/run/httpd/httpd.pid"
-D DEFAULT_SCOREBOARD="logs/apache_runtime_status"

Apache Server Information

Subpages:
Configuration Files, Server Settings, Module List, Active Hooks, Available Providers

---

Sections:
Loaded Modules, Server Settings, Startup Hooks, Request Hooks, Other Hooks, Providers

---

Loaded Modules

core.c, event.c, http_core.c, mod_access_compat.c, mod_actions.c, mod_alias.c, mod_allowmethods.c, mod_auth_basic.c, mod_auth_digest.c, mod_authn_anon.c, mod_authn_core.c, mod_authn_dbd.c, mod_authn_dbm.c, mod_authn_file.c, mod_authn_socache.c, mod_authz_core.c, mod_authz_dbd.c, mod_authz_dbm.c, mod_authz_groupfile.c, mod_authz_host.c, mod_authz_owner.c, mod_authz_user.c, mod_autoindex.c, mod_brotli.c, mod_cache.c, mod_cache_disk.c, mod_cache_socache.c, mod_cgid.c, mod_data.c, mod_dav.c, mod_dav_fs.c, mod_dav_lock.c, mod_dbd.c, mod_deflate.c, mod_dir.c, mod_dumpio.c, mod_echo.c, mod_env.c, mod_expires.c, mod_ext_filter.c, mod_filter.c, mod_headers.c, mod_http2.c, mod_include.c, mod_info.c, mod_lbmethod_bybusyness.c, mod_lbmethod_byrequests.c, mod_lbmethod_bytraffic.c, mod_lbmethod_heartbeat.c, mod_log_config.c, mod_logio.c, mod_lua.c, mod_macro.c, mod_mime.c, mod_mime_magic.c, mod_negotiation.c, mod_proxy.c, mod_proxy_ajp.c, mod_proxy_balancer.c, mod_proxy_connect.c, mod_proxy_express.c, mod_proxy_fcgi.c, mod_proxy_fdpass.c, mod_proxy_ftp.c, mod_proxy_hcheck.c, mod_proxy_http.c, mod_proxy_http2.c, mod_proxy_scgi.c, mod_proxy_uwsgi.c, mod_proxy_wstunnel.c, mod_remoteip.c, mod_reqtimeout.c, mod_request.c, mod_rewrite.c, mod_setenvif.c, mod_slotmem_plain.c, mod_slotmem_shm.c, mod_so.c, mod_socache_dbm.c, mod_socache_memcache.c, mod_socache_redis.c, mod_socache_shmcb.c, mod_status.c, mod_substitute.c, mod_suexec.c, mod_systemd.c, mod_unique_id.c, mod_unixd.c, mod_userdir.c, mod_version.c, mod_vhost_alias.c, mod_watchdog.c,

---

Server Settings

Server Version: Apache/2.4.58 (Unix)
Server Built: Jul 9 2024 11:57:06
Server loaded APR Version: 1.7.4
Compiled with APR Version: 1.7.4
Server loaded APU Version: 1.6.3
Compiled with APU Version: 1.6.3
Server loaded PCRE Version: 10.42 2022-12-11
Compiled with PCRE Version: 10.42 2022-12-11
Module Magic Number: 20120211:134
Hostname/port: 10.10.2.41:80
Timeouts: connection: 60 keep-alive: 5
MPM Name: event
MPM Information: Max Daemons: 1 Threaded: yes Forked: yes
Server Architecture: 64-bit
Server Root: /etc/httpd
Config File: /etc/httpd/conf/httpd.conf
Server Built With:
-D APR_HAS_SENDFILE
-D APR_HAS_MMAP
-D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
-D APR_USE_PROC_PTHREAD_SERIALIZE
-D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
-D APR_HAS_OTHER_CHILD
-D AP_HAVE_RELIABLE_PIPED_LOGS
-D HTTPD_ROOT="/etc/httpd"
-D SUEXEC_BIN="/usr/sbin/suexec"
-D DEFAULT_PIDLOG="/run/httpd/httpd.pid"
-D DEFAULT_SCOREBOARD="logs/apache_runtime_status"

Apache Server Information两种访问方式：

- 常规方式：http://your.server.name/server-info
- 配置阅读：http://your.server.name/server-status?config

5. 提升Apache的安全性

- 网站安全是网络安全和信息安全的重要组成部分，提升Apache网站服务器的安全性是保障网站安全的重要措施。
- Apache提供了多个手段以提升安全性，保障网站安全可靠提供服务。
  - 设置网站访问范围
  - 隐藏服务器敏感信息
  - 禁止网站目录浏览
  - 开启SELinux、防火墙进行安全防护
    - （在后续章节介绍）

5. 提升Apache的安全性

- 设置网站访问范围可以有效阻隔恶意主机攻击，极大提升网站安全性。
- Apache服务器通过Require选项实现网站访问范围限制，可通过修改配置文件实现，并支持针对网站、虚拟目录自由进行定义。

- 示例：将网站的可访问范围设置为两条规则：
  - 允许所有地址访问内容网站
  - 禁止10.10.2.116地址访问内容网站

5. 提升Apache的安全性

- 示例：将网站的可访问范围设置为两条规则：
  - 允许所有地址访问内容网站
  - 禁止10.10.2.116地址访问内容网站

- 配置文件：/etc/httpd/conf/httpd.conf
  - <Directory "/var/www/wordpress">
  - Options Indexes FollowSymLinks
  - AllowOverride None
  - #设置网站访问范围
  - <RequireAll>
  - Require all granted
  - Require not ip 10.10.2.200
  - </RequireAll>
  - </Directory>

5. 提升Apache的安全性

- Apache的Require项常用指令如下所示。
  - Require all granted，允许所有来源访问
  - Require all denied，拒绝所有来源访问
  - Require ip 127.0.0.1，只允许特定IP段访问，多个IP段之间使用空格隔开，这里是只允许IP地址为127.0.0.1的来源主机访问
  - Require host domain.com，只允许来自域名domain.com的主机访问
  - Require项可以配置多个
- Require项配合<RequireAll>、<RequireAny>、<RequireNone>标签对可以进行更加复杂的访问限制。
  - RequireAll，访问请求必须全部符合设置的允许访问规则，才能访问网站
  - RequireAny，访问请求符合设置的任意一条允许访问规则，就能访问网站
  - RequireNone，访问请求符合设置的任意一条规则，都不能访问网站。
    - 不能独立使用，一般与其他标签对配合使用

5. 提升Apache的安全性

- 隐藏 Apache 网站服务器和 PHP 解析器的敏感信息，亦可有效降低精准攻击的概率，降低服务器的风险。
- Apache网站服务器通过ServerTokens选项隐藏版本等敏感信息。
- PHP解析器通过 expose_php选项隐藏敏感信息。

- 示例：实现Apache网站服务器和PHP解析器的信息保护。
  - 修改httpd.conf文件隐藏网站服务器敏感信息
  - 修改php.ini文件隐藏PHP敏感信息

5. 提升Apache的安全性

- 示例：实现Apache网站服务器和PHP解析器的信息保护。
  - 修改httpd.conf文件隐藏网站服务器敏感信息
  - 修改php.ini文件隐藏PHP敏感信息

- 配置文件：/etc/httpd/conf/httpd.conf
  - #在文件的最底部增加下述内容，设置http头部仅返回软件名称信息
  - ServerTokens Prod

- 配置文件：/etc/php.ini
  - #修改PHP配置文件的配置项
  - expose_php = Off

5. 提升Apache的安全性

- Apache的ServerTokens共有6个选项，其作用分别如下所示。
  - ServerTokens Full，显示全部信息包含Apache支持的模块及模块版本号
  - ServerTokens Prod，仅显示网站服务器名称，即Server：Apache
  - ServerTokens Major，显示网站服务器信息包括主版本号，即Server: Apache/2
  - ServerTokens Minor，显示网站服务器信息包括次版本号，即Server: Apache/2.4
  - ServerTokens Min，显示网站服务器信息包含完整版本号，即Server: Apache/2.4.51
  - ServerTokens OS，显示网站服务器信息包含操作系统类型，即：Server: Apache/2.4.51(Unix)

- PHP的expose_php共有2个选项，其作用分别如下所示。
  - On，在网站服务器上显示已安装PHP信息
  - Off，在网站服务器上不显示已安装PHP信息

5. 提升Apache的安全性

- 禁止网站目录浏览可有效保护网站信息不被泄露，屏蔽非法用户的恶意浏览。

- 示例：禁止浏览网站目录列表。
  - 修改/etc/httpd/conf/httpd.conf配置文件禁止网站目录浏览
  - 可以针对任何一个网站、虚拟目录进行配置

5. 提升Apache的安全性

- 示例：禁止浏览网站目录列表。
  - 修改/etc/httpd/conf/httpd.conf配置文件禁止网站目录浏览
  - 可以针对任何一个网站、虚拟目录进行配置

- 配置文件：/etc/httpd/conf/httpd.conf
  - Directory "/var/www/wordpress">
  - #Options项设置为None，目录不启用任何服务器特性
  - Options None
  - AllowOverride None
  - </Directory>

5. 提升Apache的安全性

- Apache的Option常用选项如下所示。
  - Options All，显示除MultiViews之外的所有特性
  - Options MultiViews，允许多重内容被浏览
  - Options Indexes，如目录下无index文件，则显示该目录下的文件
  - Options IncludesNOEXEC，允许使用服务器端include，不可使用#exec和#include功能
  - Options Includes，允许使用服务器端include
  - Options FollowSymLinks，在目录中服务器将跟踪符号链接
  - Options SymLinksIfOwnerMatch，在目录中仅跟踪本站点内的链接
  - Options ExecCGI，在目录下准许使用CGI
  - Options后可附加多种服务器特性，特性之间使用空格隔开。

网络与信息系统智能运维课程体系学习平台

***

本课程体系由
河南中医药大学信息技术学院建设

课程体系学习平台由河南中医药大学医疗健康信息工程技术研究所开发与技术保障

网络与信息系统智能运维课程体系学习平台
https://internet.hactcm.edu.cn

互联网运维管理工程应用丛书
http://www.51xueweb.cn

扫码学习
并获取课程资源