

Chapter 3. FreeBSD 簡介

3.1. 簡介

FreeBSD 是一個類 UNIX 的作業系統。它是由一群志願者所開發的，其設計目標是提供一個穩定、可靠、且易於使用的作業系統。FreeBSD 的代碼是公開的，任何人都可以查看和修改它。FreeBSD 的安裝和配置都非常簡單，並且它支持多種硬體平台。

FreeBSD 的主要特點如下：

- 穩定可靠：FreeBSD 以其穩定性和可靠性而著稱，被廣泛用於各種關鍵任務環境。
- 性能優越：FreeBSD 具有優秀的性能，能夠處理大量的數據和請求。
- 易於使用：FreeBSD 提供了豐富的用戶界面和工具，使得安裝和配置變得非常簡單。
- 安全：FreeBSD 具有強大的安全機制，能夠有效防止各種攻擊和漏洞。
- 兼容性：FreeBSD 支持多種硬體平台，並且與許多商業軟體兼容。
- 社區支持：FreeBSD 擁有龐大的開發者和用戶社區，能夠提供及時的支持和幫助。
- 可擴展性：FreeBSD 具有強大的可擴展性，能夠根據需要添加各種功能和模塊。
- 易用性：FreeBSD 提供了豐富的用戶界面和工具，使得安裝和配置變得非常簡單。
- 穩定性：FreeBSD 以其穩定性和可靠性而著稱，被廣泛用於各種關鍵任務環境。
- 性能：FreeBSD 具有優秀的性能，能夠處理大量的數據和請求。

3.2. 安裝與配置

FreeBSD 的安裝和配置非常簡單。首先，需要選擇一個合適的硬體平台，然後下載 FreeBSD 的安裝映像。安裝過程通常包括選擇語言、時區、網絡設置等。配置過程則包括安裝必要的軟體包、設置用戶和組、以及配置系統參數。FreeBSD 提供了豐富的用戶界面和工具，使得安裝和配置變得非常簡單。

login:

[illegible]

```
FreeBSD | | | | | /etc/ttys | | | | | :
```



```
# name  getty                type status comments
#
ttyv0  "/usr/libexec/getty Pc"    xterm  on  secure
# Virtual terminals
ttyv1  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv2  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv3  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv4  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv5  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv6  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv7  "/usr/libexec/getty Pc"    xterm  on  secure
ttyv8  "/usr/X11R6/bin/xdm -nodaemon" xterm  off secure
```

我们使用 `man 8 getty` 查看 `getty` 的用法。可以看到 `getty` 的默认配置是 `secure`。
 我们使用 `man 4 tty` 查看 `tty` 的用法。可以看到 `tty` 的默认配置是 `secure`。
 我们使用 `man 8 xorg` 查看 `xorg` 的用法。可以看到 `xorg` 的默认配置是 `secure`。
 我们使用 `man 8 xdm` 查看 `xdm` 的用法。可以看到 `xdm` 的默认配置是 `secure`。

我们使用 `man 8 ttys` 查看 `ttys` 的用法。可以看到 `ttys` 的默认配置是 `secure`。

3.2.2. 配置 `secure` 和 `insecure`

FreeBSD 的 `secure` 和 `insecure` 配置是在 `/etc/ttys` 文件中定义的。
 我们使用 `man 8 ttys` 查看 `ttys` 的用法。可以看到 `ttys` 的默认配置是 `secure`。
 我们使用 `man 8 getty` 查看 `getty` 的用法。可以看到 `getty` 的默认配置是 `secure`。
 我们使用 `man 4 tty` 查看 `tty` 的用法。可以看到 `tty` 的默认配置是 `secure`。
 我们使用 `man 8 xorg` 查看 `xorg` 的用法。可以看到 `xorg` 的默认配置是 `secure`。
 我们使用 `man 8 xdm` 查看 `xdm` 的用法。可以看到 `xdm` 的默认配置是 `secure`。

我们使用 `man 8 ttys` 查看 `ttys` 的用法。可以看到 `ttys` 的默认配置是 `secure`。

```
# name  getty                type status comments
#
# If console is marked "insecure", then init will ask for the root password
# when going to single-user mode.
console none                unknown off secure
```

我们使用 `man 8 ttys` 查看 `ttys` 的用法。可以看到 `ttys` 的默认配置是 `secure`。
 我们使用 `man 8 getty` 查看 `getty` 的用法。可以看到 `getty` 的默认配置是 `secure`。
 我们使用 `man 4 tty` 查看 `tty` 的用法。可以看到 `tty` 的默认配置是 `secure`。
 我们使用 `man 8 xorg` 查看 `xorg` 的用法。可以看到 `xorg` 的默认配置是 `secure`。
 我们使用 `man 8 xdm` 查看 `xdm` 的用法。可以看到 `xdm` 的默认配置是 `secure`。

root 的权限，并安装必要的软件包。

Note

在 `insecure` 模式下，系统会提示输入密码。在 FreeBSD 中，可以通过设置 `secure` 来避免这种情况。

3.2.3. 安装 VESA 驱动程序

FreeBSD 提供了 VESA 驱动程序，支持 1024x768, 1280x1024 分辨率。安装步骤如下：

```
# kldload vesa
```

安装完成后，可以通过 `vidcontrol(1)` 来设置显示模式。具体操作如下：

```
# vidcontrol -i mode
```

在提示符下输入 `vidcontrol(1)` 并按回车键，即可进入交互式模式。此时可以输入 `mode` 来查看支持的分辨率。

```
# vidcontrol MODE_279
```

最后，需要编辑 `/etc/rc.conf` 文件，添加以下配置项：

```
allscreens_flags="MODE_279"
```

3.3. 安装 X11 窗口系统

FreeBSD 提供了 X11 窗口系统，支持多种图形界面。安装步骤如下：

1. 安装 X11 窗口系统：

- FreeBSD 提供了多种 X11 窗口系统，包括 Xorg 和 XFree86。
- 安装 X11 窗口系统后，需要配置 X11 窗口系统。

- 在 `/etc/passwd` 文件中添加新的用户信息。
- 在 `/etc/group` 文件中添加新的组信息。

3.3.1. 用户管理

FreeBSD 提供了多种用户管理工具，包括 `adduser`、`deluser`、`passwd` 等。

在添加用户时，需要指定用户名、密码、组名等信息。

3.3.1.1. 添加用户

使用 `adduser` 命令可以添加新用户。该命令会交互式地提示用户输入用户名、密码、组名等信息。

在添加用户时，可以选择将用户添加到已有的组，或者创建一个新的组。

`nobody` 是一个特殊的用户，通常用于运行不需要权限的服务。该用户的密码为空。

3.3.1.2. 删除用户

使用 `deluser` 命令可以删除用户。该命令会提示用户输入用户名，并确认删除。

在删除用户时，可以选择是否删除用户的家目录。

FreeBSD 还提供了 `useradd` 命令，用于添加用户。

User name

```
login: root
passwd:
group: root
full name: root
home directory: /root
shell: /bin/csh
is system user: no
is login user: yes
is locked: no
is inactive: no
is non-unique: no
is unique: yes
is valid: yes
is valid until: 8 days
```

Password

在添加用户时，需要指定密码。

User ID (UID)

UID (UID) FreeBSD 系统 用户 用户 用户 用户 用户 . 用户
用户 用户 用户 用户 用户 用户 用户 UID 用户 . 用户 用户 用户 用户
用户 用户 用户 用户 用户 65535 用户 UID 用户 用户 用户 .

Group ID (GID)

用户 ID (GID) 用户 用户 用户 用户 用户 用户 用户 用户 . 用户
用户 UID 用户 GID 用户 用户 用户 用户 用户 用户 . 用户 用户
用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 . GID
用户 用户 用户 用户 用户 用户 65535 用户 GID 用户 用户 用户 .

Login class

用户 用户 用户 用户 用户 , 用户 用户 用户 用户 用户 用户
用户 用户 . 用户 用户 用户 用户 用户 用户 用户 用户
用户 .

Password change time

用户 用户 用户 用户 . 用户 用户 用户 用户 用户 用户 用户
用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 .

Account expiration time

用户 FreeBSD 用户 用户 用户 . 用户 用户 用户 用户 用户 用户 用户
用户 用户 pw(8) 用户 用户 用户 用户 . 用户 用户 用户 用户
用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 .

User's full name

用户 用户 FreeBSD 用户 用户 用户 用户 用户 用户 用户 用户 用户 .
用户 用户 用户 用户 , 用户 用户 用户 8 用户 用户 .

Home directory

用户 用户 用户 用户 用户 用户 . 用户 用户 用户 用户 用户
用户 . 用户 用户 用户 用户 用户 /home/username 用户
/usr/home/username 用户 用户 用户 . 用户 用户 用户 用户 用户 用户
用户 用户 .

User shell

用户 用户 用户 用户 用户 用户 用户 用户 . 用户 用户 用户 用户 用户
用户 用户 用户 用户 用户 用户 用户 用户 用户 用户 .

3.3.1.3. 用户 用户

root 的权限，可以执行任何操作。但是，root 用户不能直接安装软件包。这是因为 root 用户没有安装权限。为了安装软件包，我们需要使用 su 命令切换到 root 用户，然后使用 make install 命令安装软件包。最后，使用 exit 命令退出 root 用户。

在 su 命令后面加上 (1) 可以指定要切换到的用户。例如，su(1) 表示切换到 root 用户。在 su 命令后面加上 wheel 可以指定要加入的用户组。例如，su wheel 表示切换到 wheel 用户组。在 su 命令后面加上 root 可以指定要切换到的用户。例如，su root 表示切换到 root 用户。

在 make install 命令后面加上 -f 可以强制安装。例如，make install -f 表示强制安装。在 exit 命令后面加上 ! 可以强制退出。例如，exit ! 表示强制退出。

Example 1. 安装软件包

```
% configure
% make
% su -
Password:
# make install
# exit
%
```

在 su 命令后面加上 (1) 可以指定要切换到的用户。例如，su(1) 表示切换到 root 用户。在 su 命令后面加上 wheel 可以指定要加入的用户组。例如，su wheel 表示切换到 wheel 用户组。在 su 命令后面加上 root 可以指定要切换到的用户。例如，su root 表示切换到 root 用户。

3.3.2. 安装软件包

FreeBSD 的包管理器是 pkg 命令。它可以帮助我们安装和卸载软件包。在 pkg 命令后面加上 install 可以安装软件包。例如，pkg install 表示安装软件包。在 pkg 命令后面加上 remove 可以卸载软件包。例如，pkg remove 表示卸载软件包。

Table 1. 安装软件包的命令

Command	Summary
---------	---------


```
Enter password again:
Lock out the account after creation? [no]:
Username   : jru
Password   : ****
Full Name   : J. Random User
Uid        : 1001
Class      :
Groups     : jru wheel
Home       : /home/jru
Shell      : /usr/local/bin/zsh
Locked     : no
OK? (yes/no): yes
adduser: INFO: Successfully added (jru) to the user database.
Add another user? (yes/no): no
Goodbye!
```

Note

adduser(8) is a program that adds a new user to the system. It is part of the base system and is located in the /usr/sbin directory. It is a shell script that prompts the user for information and then adds the user to the system.

3.3.2.2. Adding a user

The adduser(8) program is used to add a new user to the system. It is a shell script that prompts the user for information and then adds the user to the system. The program is located in the /usr/sbin directory.

1. The [crontab\(1\)](#) program is used to edit the crontab file. It is a shell script that prompts the user for information and then adds the user to the system.
2. The [at\(1\)](#) program is used to schedule a command to be executed at a specific time. It is a shell script that prompts the user for information and then adds the user to the system.
3. The [SIGKILL](#) signal is used to kill a process. It is a signal that is sent to the process and causes it to terminate.
4. The [mail](#) command is used to send an email. It is a command that is used to send an email to a specific address.
5. The [mail](#) command is used to send an email. It is a command that is used to send an email to a specific address.
6. The [/var/mail](#) directory is used to store mailboxes. It is a directory that is used to store mailboxes for each user.
7. The [/tmp](#), [/var/tmp](#) and [/var/tmp/vi.recover](#) directories are used to store temporary files. They are directories that are used to store temporary files.
8. The [/etc/group](#) file is used to store group information. It is a file that is used to store group information for each user.

rmuser(8) 命令如下所示。

9. 使用 rmuser 命令删除用户 jru，并删除其主目录。

`rmuser(8)` 命令如下所示。

使用 rmuser 命令删除用户 jru。

Example 3. rmuser 命令

```
# rmuser jru
```

命令输出如下所示：

```
Matching password entry:
jru*:1001:1001::0:0:j. Random User:/home/jru:/usr/local/bin/zsh
Is this the entry you wish to remove? y
Remove user's home directory (/home/jru)? y
Removing user (jru): mailspool home passwd.
```

3.3.2.3. 使用 chpass 命令

使用 chpass(1) 命令更改用户 jru 的密码。

使用 chpass 命令更改用户 jru 的密码。

Note

使用 chpass 命令更改用户 jru 的密码。

使用 chpass 命令更改用户 jru 的密码。

Example 4. 使用 chpass 命令

```
# chpass
```

命令输出如下所示：

Other information:

Other information:

Note



















1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200
 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300
 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400
 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500
 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600
 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700
 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800
 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900
 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000
 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 103

Revision #6

Created 21 April 2024 03:59:53 by (MeatDumpling)

Updated 17 September 2024 02:55:47 by (MeatDumpling)