

4.1 SysV-init システムの起動をカスタマイズする

このトピックでは、Linux システムの起動の流れとサービスの起動について出題されます。

● 例題

システム起動時のサービスの自動起動設定を行うコマンドを選択して下さい。

- A) service
- B) chkconfig
- C) telinit
- D) chkserv

解答： B

● 概要 (試験範囲から抜粋)

重要度	3
さまざまなランレベルのシステムサービスの動作を照会および変更する。init の構造とブートプロセスについての完全な理解が必要。これには、ランレベルの操作も含まれる。	
主な知識範囲	
Linux Standard Base Specification (LSB) SysV init の環境	
重要なファイル、用語、ユーティリティ	
/etc/inittab /etc/init.d/ /etc/rc.d/ chkconfig update-rc.d init と telinit	

4.1.1 ブートプロセス

マシンの電源を入れてから Linux システムが起動するまでの手順を以下に示します。

1. BIOS が起動し、ハードウェアの初期化とブートデバイスの決定を行う。
2. BIOS がブートデバイス (一般的にはハードディスク) の MBR から GRUB や LILO といったブートローダを読み出す。
3. ブートローダがカーネルを読み込む。
4. カーネルが起動する。カーネルは、イニシャルRAMディスクを暫定的な root ファイルシステムとしてマウントし、各種デバイスの初期化および実際の root ファイルシステムをマウントするために必要なドライバをロードし、root ファイルシステムをマウントする。
5. init プログラム (プロセス ID: 1) を実行する。
6. init プログラムは/etc/inittab の設定に従い、ランレベルの決定、初期化スクリプト (/etc/rc.d/rc.sysinit) を実行する。
7. /etc/rc.d/rc スクリプトを実行し、各種サービスが起動する。
8. ログインプロンプトを表示する。

4.1.2 /etc/inittab

/etc/inittab ファイルは、すべてのプロセスの親である init プロセスの設定ファイルです。

/etc/inittab (コメント一部省略)

```
#
# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left. Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6

# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon
```

/etc/inittab の書式

```
id:runlevels:action:process
```

- id
エントリの識別子。
- runlevels
ランレベルの指定。指定されたランレベルで動作する際に実行される。
「2345」のように複数同時に指定できる。この場合ランレベル 2～5 で実行される。
- action
どのような動作を表すかを指定する。action に指定できる主な項目を示す。

action	意味
respawn	process に指定されたプロセスを起動し、プロセスが終了した場合は常に再起動する。
wait	指定したランレベルになった時に process に指定されたプロセスを起動し、終了をまつ。
once	指定したランレベルになった時に一度だけ実行する。
initdefault	デフォルトのランレベルを決定する。process の欄は無視される。
sysinit	システムのブート中に実行する。runlevels の欄は無視される。
powerfail	電源に異常が起きた時に実行される。電源異常は、UPS と通信しているプロセスから知られる。
ctrlaltdel	[Ctrl] + [Alt] + [Delete] キーが押された場合に実行される。

- process
実行されるプロセスを指定する。

■ /etc/inittab の主な内容

1. ランレベルの決定

/etc/inittab (抜粋)

```
id:5:initdefault:
```

起動時のデフォルトのランレベルを設定します。

2. 初期化処理の実行

/etc/inittab (抜粋)

```
si::sysinit:/etc/rc.d/rc.sysinit
```

初期化スクリプト `/etc/rc.d/rc.sysinit` を実行します。action に `sysinit` が指定されているためシステムのブート時に実行されます。主に以下の処理を実行します。

- ホスト名の設定
- `/proc` ファイルシステムのマウント
- ルートファイルシステムのチェック
- RAID、LVM の有効化
- ルートファイルシステムの書き込み可能での再マウント
- 他のファイルシステムのチェック
- 他のファイルシステムのマウント
- quota の有効化
- swap の有効化

なお、ファイルシステムのチェック・マウントは、`/etc/fstab` ファイルに従って行われます。

3. 各種サービスの起動

/etc/inittab (抜粋)

```
l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6
```

/etc/rc.d/rc スクリプトにより、指定されたランレベルに不要なサービスを停止し、必要なサービスを起動します。action に wait が指定されているため、init は /etc/rc.d/rc スクリプトの終了を待ち合わせます。

4. ログインプロンプトの表示

/etc/inittab (抜粋)

```
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

ランレベル 2 から 5 のときに、/sbin/mingetty を起動し、ログインプロンプトを表示します。action に respawn が指定されているため、ログアウトにより mingetty が終了した後 init は再度 mingetty を起動し、ログインプロンプトを表示します。

/etc/inittab (抜粋)

```
# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon
```

ランレベル 5 のときに、/etc/X11/prefdm を実行し、X Window System のログイン画面を表示します。

4.1.3 システムサービスの動作のカスタマイズ

/sbin/init は、/etc/inittab ファイルの設定に従い、/etc/rc.d/rc スクリプトを実行しサービスを起動します。

/etc/rc.d/rc (抜粋)

```
# First, run the KILL scripts.
for i in /etc/rc$runlevel.d/K* ; do
    . . .

    action $"Stopping $sysys: " $i stop
    . . .
done
# Now run the START scripts.
for i in /etc/rc$runlevel.d/S* ; do
    . . .

    action $"Starting $sysys: " $i start
    . . .
done
```

/etc/rc.d/rc スクリプトでは、各ランレベルに不要なサービスを停止するために、/etc/rcX.d ディレクトリ内にある K で始まるスクリプトファイルを、引数に stop を指定し実行することでサービスを停止します。その後、/etc/rcX.d ディレクトリ内にある S で始まるスクリプトファイルを、引数に start を指定し実行することでサービスを起動します。

したがって、/etc/rcX.d ディレクトリ内のファイルを調べることで、どのようなサービスを起動・停止するのかを把握することができます。

```
# ls /etc/rc3.d
K02NetworkManager  K73ypbind          S09isdn             S56rawdevices
K05saslauthd        K74nscd            S09pcmcia           S56xinetd
K10dc_server        K74ntpd            S10network          S80sendmail
. . .
K45named            K99readahead       S40smartd           S99local
K50netdump           K99readahead_early S44acpid
K50tux              S06cpuspeed        S55cups
K50vsftpd           S08iptables        S55sshd
```

また、`/etc/rcX.d` ディレクトリ内のファイルは、`/etc/init.d` ディレクトリ内の起動スクリプトへのシンボリックリンクになっています。

```
# ls /etc/rc?.d/*sshd
/etc/rc0.d/K25sshd /etc/rc2.d/S55sshd /etc/rc4.d/S55sshd /etc/rc6.d/K25sshd
/etc/rc1.d/K25sshd /etc/rc3.d/S55sshd /etc/rc5.d/S55sshd

# ls -l /etc/rc?.d/*sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc0.d/K25sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc1.d/K25sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc2.d/S55sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc3.d/S55sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc4.d/S55sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc5.d/S55sshd -> ../init.d/sshd
lrwxrwxrwx 1 root root 14 1月 17 16:59 /etc/rc6.d/K25sshd -> ../init.d/sshd
```

`/etc/init.d` ディレクトリ内には、各サービスを制御する起動スクリプトがあります。各スクリプトは、シェルスクリプトファイルであり、実行時の引数を元に処理が分岐するように作られています。

```
# ls /etc/init.d/
FreeWnn      dhcrelay lisa      portmap squid
NetworkManager diskdump lm_sensors postfix sshd
acpid        dund      mailman postgresql syslog
...

# /etc/init.d/sshd status
sshd (pid 3071 1971) を実行中...

# /etc/init.d/sshd stop
sshd を停止中: [ OK ]

# /etc/init.d/sshd status
sshd は停止しています

# /etc/init.d/sshd start
sshd を起動中: [ OK ]

# /etc/init.d/sshd status
sshd (pid 3150 3071) を実行中...
```


/etc/init.d/sshd (抜粋)

```
#!/bin/bash
#
# Init file for OpenSSH server daemon
. . .
start()
{
    # Create keys if necessary
    do_rsa1_keygen
    do_rsa_keygen
    do_dsa_keygen

    echo -n "$Starting $prog:"
    initlog -c "$SSHD $OPTIONS" && success || failure
    RETVAL=$?
    [ "$RETVAL" = 0 ] && touch /var/lock/subsys/sshd
    echo
}

stop()
{
    echo -n "$Stopping $prog:"
    killproc $SSHD -TERM
    RETVAL=$?
    [ "$RETVAL" = 0 ] && rm -f /var/lock/subsys/sshd
    echo
}

. . .
case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    reload)
        reload
        ;;
    status)
        status $SSHD
        RETVAL=$?
        ;;
    . . .
    *)
        echo $"Usage: $0 {start|stop|restart|reload|condrestart|status}"
        RETVAL=1
esac
```

このスクリプトを実行するシェルの指定

\$1 は、実行時の第 1 引数を表します。
引数により処理が分岐します。

独自のサービスを起動したい場合には、/etc/rcX.d ディレクトリ内に起動スクリプトを配置することでランレベルごとにサービスの自動起動を設定することができます。通常は/etc/init.d ディレクトリに起動スクリプトを作成し、/etc/rcX.d ディレクトリ内に S や K で始まるシンボリックリンクを作成します。

```
# ls /etc/init.d/customservice
/etc/init.d/customservice
# ln -s /etc/init.d/customservice /etc/rc3.d/S80customservice
```

ln コマンドによるシンボリックリンクの作成は、動作順序¹²の確認など作業が煩雑になるため、一般には ln コマンドを用いずにディストリビューションに同梱されるコマンドを使用します。

RedHat 系ディストリビューション	chkconfig コマンド
Debian 系ディストリビューション	update-rc.d コマンド

chkconfig		
機能	サービス起動の設定/確認を行う。	
書式	chkconfig [<u>option</u>] <u>service</u> シンボリックリンク配置、削除 chkconfig <u>service</u> on off サービスの起動設定	
主なオプション	--list	サービス起動の設定を確認する。
	--add	シンボリックリンク (/etc/rcX.d/*service) を作成する。
	--del	シンボリックリンクを削除する。
<u>service</u> には、/etc/init.d ディレクトリに配置した起動スクリプトファイル名を指定する。		

```
# chkconfig --list sshd
sshd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
# ls /etc/rc?.d/*sshd
/etc/rc0.d/K25sshd /etc/rc2.d/S55sshd /etc/rc4.d/S55sshd /etc/rc6.d/K25sshd
/etc/rc1.d/K25sshd /etc/rc3.d/S55sshd /etc/rc5.d/S55sshd
# chkconfig --del sshd
# ls /etc/rc?.d/*sshd
ls: /etc/rc?.d/*sshd: そのようなファイルやディレクトリはありません
# chkconfig --add sshd
# ls /etc/rc?.d/*sshd
/etc/rc0.d/K25sshd /etc/rc2.d/S55sshd /etc/rc4.d/S55sshd /etc/rc6.d/K25sshd
/etc/rc1.d/K25sshd /etc/rc3.d/S55sshd /etc/rc5.d/S55sshd
```

¹² 動作順序は、S、K に続く 2 桁の数字が小さい順になります。

update-rc.d	
機能	サービス起動の設定を行う。
書式	<pre>update-rc.d <u>service</u> defaults [<u>NN</u>]</pre> サービスの初期設定に従ってシンボリックリンク (/etc/rcX.d/*<u>service</u>) を作成する <pre>update-rc.d -f <u>service</u> remove</pre> シンボリックリンクを削除する <pre>update-rc.d <u>service</u> start stop <u>NN</u> <u>level</u>...</pre> ランレベルごとにサービスの起動設定を行う
<u>service</u> には、/etc/init.d ディレクトリに配置した起動スクリプトファイル名を指定する。	

```

debian:~# ls /etc/rc?.d/*ssh
/etc/rc0.d/K20ssh /etc/rc2.d/S20ssh /etc/rc4.d/S20ssh /etc/rc6.d/K20ssh
/etc/rc1.d/K20ssh /etc/rc3.d/S20ssh /etc/rc5.d/S20ssh
debian:~# update-rc.d -f ssh remove
Removing any system startup links for /etc/init.d/ssh ...
  /etc/rc0.d/K20ssh
  /etc/rc1.d/K20ssh
  /etc/rc2.d/S20ssh
  /etc/rc3.d/S20ssh
  /etc/rc4.d/S20ssh
  /etc/rc5.d/S20ssh
  /etc/rc6.d/K20ssh
debian:~# ls /etc/rc?.d/*ssh
ls: cannot access /etc/rc?.d/*ssh: そのようなファイルやディレクトリはありません
debian:~# update-rc.d ssh defaults
Adding system startup for /etc/init.d/ssh ...
  /etc/rc0.d/K20ssh -> ../init.d/ssh
  /etc/rc1.d/K20ssh -> ../init.d/ssh
  /etc/rc6.d/K20ssh -> ../init.d/ssh
  /etc/rc2.d/S20ssh -> ../init.d/ssh
  /etc/rc3.d/S20ssh -> ../init.d/ssh
  /etc/rc4.d/S20ssh -> ../init.d/ssh
  /etc/rc5.d/S20ssh -> ../init.d/ssh
debian:~# ls /etc/rc?.d/*ssh
/etc/rc0.d/K20ssh /etc/rc2.d/S20ssh /etc/rc4.d/S20ssh /etc/rc6.d/K20ssh
/etc/rc1.d/K20ssh /etc/rc3.d/S20ssh /etc/rc5.d/S20ssh
debian:~#

```