

SCRIPT AVVIO SLACKWARE 15

Ecco una piccola analisi sugli script di avvio della distribuzione slackware 15. Non si tratta di un'analisi esaustiva, ma solo di una piccola panoramica. Se avro' tempo e voglia, in futuro aggiungero' altri dettagli, magari con esempi di configurazione.

- init.d/ script di avvio per chi desidera usare lo stile tipico di linux. Ricordiamo che slackware usa lo stile di bsd, con i file di avvio/arresto locati in /etc/rc.d ricordiamo inoltre che slackware non usa systemd , non usa upstart e non usa nessuna delle modernita' delle altre distro
- rc.0 invocato allo spegnimento del sistema
- rc.4 invocato all'avvio di slackware in modalita' grafica. Consultare anche il file /etc/inittab
- rc.6 invocato al riavvio del sistema. e' uno script che si occupa di stoppare i servizi , esegue il logout delle utenze e poi riavvia il sistema
- rc.K abbreviazione di rc.Kill . viene invocato all'arresto del sistema. In realta' lo scopo dello script e' soprattutto chiudere in modo corretto ed ordinato tutti i servizi, eseguire la pulizia dei terminali e poco altro.
- rc.M viene eseguito all'avvio del sistema in runlevel 3 , che e' il runlevel predefinito in slackware. si occupa di attivare tutti i servizi e prepara il sistema per l'utilizzo quotidiano.
- rc.S script di inizializzazione eseguito all'avvio del sistema. serve a fare il mount dei filesystem, verifica integrita' dei filesystem, impostazione orologio di sistema, caricamento moduli del kernel, inizializzazione della rete, mount delle zone LUKS (cryptsetup), attivazione dello swap ed altro.
- rc.acpid si occupa della gestione dell'alimentazione. gestione dei tasti speciali, sospensione, ibernazione e stato di batteria dei portatili

rc.alsa si occupa della gestione del sonoro, caricando i moduli corretti per la scheda audio

rc.alsa-oss garantisce la compatibilita' tra ALSA e OSS .

rc.atalk si occupa dell'avvio/stop/restart del demone AppleTalk, cioe' il protocollo apple di condivisione e supporto stampanti

rc.atd avvia il demone at . si tratta della gestione dei comandi at , cioe' l'esecuzione di un comando nel futuro.

rc.autofs Avvia il servizio autofs: Lo script avvia il demone autofs, che si occupa di montare automaticamente le risorse (come condivisioni di rete o dispositivi esterni) quando vengono accedute. In altre parole, autofs monta i file system solo quando un utente o un processo cerca di accedervi, e li smonta quando non sono più necessari.

rc.bind avvia/ferma/riavvia il servizio di risoluzione dei nomi

rc.bluetooth avvia/riavvia/ferma il servizio bluetooth. carica i moduli necessari per gestire le periferiche bluetooth, inoltre si occupa delle comunicazioni tra i dispositivi bluetooth

rc.cgconfig Lo script rc.cgconfig viene utilizzato per avviare e configurare il sistema di cgroups all'avvio del sistema. Questo script fa parte del processo di gestione dei cgroups v1 (nella versione classica) o cgroups v2 (in sistemi più recenti), che permette di gestire e organizzare i processi in gruppi con limiti e priorità specifici per l'utilizzo delle risorse del sistema. Di fatto si occupa del risparmio energetico

rc.cgred Serve ad avviare, fermare o riavviare il demone cgrulesengd, che fa parte del pacchetto libcgroup. Questo demone è responsabile dell'applicazione automatica delle regole ai processi in base agli utenti o ai gruppi, usando i Control Groups (cgroups) di Linux. In breve: rc.cgred è lo script di servizio. cgrulesengd è il demone

che applica le regole di cgroups definite in /etc/cgrules.conf

- rc.cpubfreq è uno script di avvio (init) che serve per configurare la frequenza della CPU al boot del sistema, tramite il governor cpufreq del kernel Linux. Serve a impostare il governor della CPU, ovvero la politica con cui il kernel decide a che velocità deve lavorare la CPU in base al carico di lavoro, consumo energetico, e prestazioni.
- rc.crond cron d'è il demone che si occupa di eseguire i comandi programmati ad intervalli regolari. di fatto è la programmazione di attività che devono essere eseguite in modo ripetitivo e prolungato nel tempo
- rc.cups si tratta del sistema di stampa. consente di stampare su stampanti locali o di rete, aggiungere/eliminare stampanti, gestire le code di stampa.
- rc.cups-browsed avvia/ferma/riavvia il servizio cups-browsed, il quale si occupa di auto-rilevare le stampanti remote, supporto nel protocollo avahi o ldap
- rc.dnsmasq avvia/ferma/riavvia il servizio dnsmasq. dnsmasq è un server leggero che può fare da: Server DNS locale (cache e forwarding DNS), Server DHCP (assegnazione IP in LAN), Server TFTP (facoltativamente) Supporta anche PXE boot per avvio da rete
- rc.dovecot script di avvio/stop/restart per dovecot. Dovecot è un Mail Delivery Agent (MDA), ovvero: Fornisce accesso alla posta elettronica tramite IMAP e/o POP3 Supporta vari formati di mailbox (Maildir, mbox, ecc.) Lavora insieme a un MTA (tipo Postfix, Sendmail, Exim) per permettere agli utenti di leggere le email Ha supporto per autenticazione sicura, SSL/TLS, e filtri Sieve
- rc.elogind Il file /etc/rc.d/rc.elogind in Slackware 15 è uno script di init usato per avviare, fermare e gestire il servizio elogind durante il processo di boot o di shutdown del sistema. elogind è un fork di logind (originariamente parte di systemd), progettato per essere usato in sistemi che non utilizzano systemd, come appunto Slackware. Fornisce funzionalità legate alla gestione delle sessioni utente, come: gestione dell'accesso al login, sospensione e ripresa del sistema, gestione dei dispositivi per utente, controllo di chi ha il "seating" attivo (console, X session, ecc.).
- rc.font Il file /etc/rc.d/rc.font in Slackware 15 è uno script di init che viene usato per impostare il font della console testuale (cioè il font che vedi quando sei fuori da X, in modalità testo, tipo su tty1-tty6).

rc.fuse script di init usato per caricare il modulo del kernel FUSE (Filesystem in Userspace) e avviare i servizi necessari per usare filesystem gestiti in spazio utente. Cos'è FUSE? FUSE (Filesystem in Userspace) è una tecnologia che permette agli utenti non-root di montare e usare filesystem senza dover scrivere codice nel kernel. Viene usata per filesystem come: NTFS-3G (lettura/scrittura di partizioni NTFS), SSHFS (filesystem via SSH), EncFS, UnionFS, rclone, e tanti altri.

rc.fuse3 avvia/ferma/riavvia il servizio fuse 3, che è una versione più moderna e separata di FUSE (Filesystem in Userspace).

rc.gpm script di init che serve ad avviare, fermare o riavviare il demone gpm, che gestisce il mouse nella console testuale (cioè fuori da X).

rc.haveged fornisce entropia al kernel per migliorare la generazione di numeri casuali, specialmente su sistemi dove l'entropia iniziale è bassa (es. VM, server headless, ecc.).

rc.httpd avvia/ferma/riavvia il server apache. in pratica è il web server che eroga le pagine web ai client richiedenti.

rc.icecc-scheduler avvia/ferma/riavvia il demone Icecream Scheduler, che funge da punto centrale in una rete di compilazione Icecream. Il suo compito principale è distribuire i compiti di compilazione tra i nodi disponibili e fornire dati per i monitor. In pratica, coordina la distribuzione delle attività di compilazione tra i vari nodi del sistema, ottimizzando l'uso delle risorse disponibili.

rc.iceccd script di init per avviare, fermare o gestire il demone iceccd → cioè il compilatore distribuito del sistema Icecream.

rc.icecream.conf script di configurazione associato al sistema di compilazione distribuita Icecream, usato per impostare parametri che saranno poi letti dagli script di avvio rc.iceccd e rc.icecc-scheduler.

rc.inet1 configura le interfacce di rete durante il processo di avvio, prima di avviare qualsiasi altro servizio di rete (come server DNS, HTTP, ecc.). Questo script è parte integrante della gestione della rete

in Slackware.

- rc.inet1.conf È il file di configurazione delle interfacce di rete tradizionali in Slackware, usato dallo script /etc/rc.d/rc.inet1 per configurare la rete al boot, se non si vuole usare NetworkManager.
- rc.inet2 script di avvio eseguito da /etc/rc.d/rc.M (lo script principale del runlevel multiutente in Slackware). Viene eseguito dopo che le interfacce di rete sono state configurate
- rc.inetd script di avvio per il demone inetd, anche conosciuto come super-server.
- rc.ip_forward serve ad abilitare o disabilitare il forwarding IP, ovvero la capacità del sistema di inoltrare pacchetti di rete tra più interfacce – funzionalità tipica di un router
- rc.kadmind serve ad avviare, fermare o riavviare il demone kadmind, che fa parte del sistema Kerberos. Più precisamente, kadmind è il Kerberos administration daemon – ovvero il servizio che permette la gestione centralizzata degli account e dei ticket Kerberos.
- rc.keymap script di init usato per impostare la mappatura della tastiera (keymap) durante l'avvio del sistema. così facendo è possibile avere la mappatura della tastiera in Italiano ancora prima di avviare l'ambiente grafico
- rc.kpropd si tratta del sistema kerberos, ed è usato per avviare il demone kpropd. Si tratta di un sistema di autenticazione via rete. serve per sincronizzare i database dei ticket kerberos con uno o più'
- rc.krb5kdc script di avvio legato al sistema di autenticazione Kerberos, e serve per gestire il demone principale del KDC (Key Distribution Center), cioè krb5kdc
- rc.local a disposizione dell'admin. è l'ultimo file a venire eseguito nella sequenza di accensione. i comandi qui dentro inseriti vengono eseguiti in sequenza. Serve per aggiungere comandi personalizzati.
- rc.loop script legato alla gestione dei dispositivi loopback, ovvero i loop device. Lo script rc.loop è usato per caricare automaticamente il

modulo del kernel loop (cioè loop.ko) durante il boot del sistema, se è necessario.

- rc.lxc script di avvio (init script) che gestisce LXC containers nel sistema
- rc.mcelog usato per gestire il demone mcelog, che serve per monitorare errori hardware a basso livello, in particolare Machine Check Exceptions (MCE)
- rc.messagebus serve per gestire l'avvio e lo stop del D-Bus system daemon, ovvero dbus-daemon -system È essenziale per il corretto funzionamento di ambienti desktop come KDE, Xfce, GNOME (se installato manualmente), e anche per alcune applicazioni grafiche
- rc.modules serve a caricare manualmente i moduli del kernel all'avvio del sistema, prima che partano altri servizi.
- rc.modules.local file opzionale, pensato per contenere personalizzazioni locali relative al caricamento dei moduli del kernel, senza modificare il file principale rc.modules Slackware segue la filosofia "non toccare i file di sistema se puoi evitarlo". Quindi: rc.modules è quello fornito dal sistema, può essere sovrascritto durante aggiornamenti. rc.modules.local è il tuo spazio personale dove mettere i moduli che vuoi caricare manualmente
- rc.mysqlld script di avvio e gestione del server MySQL (o MariaDB), ed è parte integrante del sistema se si usa un database SQL. in realta' avvia/ferma/riavvia mariadb , che e' un fork di mysql
- rc.networkmanager script di avvio del demone NetworkManager, che è uno dei metodi moderni per gestire connessioni di rete (sia cablate che wireless)
- rc.nfsd script di avvio del server NFS (Network File System), ed è fondamentale se vuoi condividere directory via rete tra sistemi Unix/Linux
- rc.nss-pam-ldapd serve per avviare il demone nslcd, che è parte del sistema nss-pam-ldapd, usato per integrare LDAP con l'autenticazione del sistema
- rc.ntpd serve per avviare, fermare o riavviare il demone ntpd, cioè il

Network Time Protocol Daemon, che mantiene l'orologio del sistema sincronizzato con i server NTP in rete

- rc.openldap serve per avviare, fermare o riavviare il server LDAP fornito da OpenLDAP, ovvero il demone slapd.
OpenLDAP è una suite open source che implementa il protocollo LDAP (Lightweight Directory Access Protocol),
usato per: Gestire un database centralizzato di utenti, gruppi, permessi, host, certificati, ecc.
Fare autenticazione centralizzata (con PAM, NSS)
Costruire directory aziendali o backend per login, email, Samba, ecc. Il server principale di OpenLDAP è slapd (Standalone LDAP Daemon)
- rc.openvpn serve per avviare, fermare o riavviare il servizio OpenVPN, un software che implementa una VPN (Virtual Private Network) per creare una connessione sicura tra computer o reti su internet
- rc.pcmcia script di avvio utilizzato per gestire il supporto alle schede PCMCIA (o PC Cards), che sono quelle vecchie schede di espansione usate soprattutto nei laptop anni '90 e primi 2000 (tipo modem, schede di rete, ecc.)
- rc.php-fpm script di avvio pensato per gestire il demone PHP-FPM. PHP-FPM (FastCGI Process Manager) è una versione avanzata di PHP progettata per l'esecuzione come demone in background, tipicamente usata in ambienti web con Nginx, Apache (con mod_proxy_fcgi), o altri server che parlano FastCGI
- rc.postfix avvia/ferma/riavvia il serve di posta postfix
- rc.pulseaudio avvia/ferma/riavvia il demone che gestisce il sistema sonoro
- rc.rpc script di avvio che serve a gestire i servizi RPC (Remote Procedure Call), utilizzati in particolare da NFS (Network File System), cioè per la condivisione di file su rete.
- rc.samba avvia/ferma/riavvia il servizio samba, cioè il sistema di condivisione files e stampanti con i sistemi windows
- rc.saslauthd demone di autenticazione del sistema Cyrus SASL (Simple Authentication and Security Layer).
- rc.serial script di avvio per configurare manualmente le porte seriali del sistema.
- rc.setterm script di avvio che si occupa di configurare il comportamento del terminale di sistema (console virtuale) durante il boot. Usa il

comando setterm, che serve a modificare vari aspetti della console testuale (non X11), come:
lo spegnimento automatico dello schermo (blanking)
la durata del blanking
l'accensione/spegnimento dello screensaver
la colorazione del testo
la gestione del cursore

- rc.smartd serve per gestire il demone smartd, che fa parte del pacchetto smartmontools. Questo demone è usato per monitorare lo stato di salute dei dischi rigidi e SSD attraverso il sistema S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology).
- rc.snmpd serve per gestire il demone SNMP (snmpd), che fa parte del pacchetto net-snmp. In breve: snmpd è il demone SNMP (Simple Network Management Protocol) che consente a un sistema di essere monitorato da software di gestione di rete (come Zabbix, Nagios, Cacti, o altri strumenti SNMP-based).
- rc.sshd avvia/ferma/riavvia il servizio ssh, cioè la connessione cifrata.
- rc.syslog serve per gestire il sistema di logging, ovvero per avviare il demone syslog (solitamente syslogd e klogd), che è responsabile della raccolta e gestione dei messaggi di log del sistema.
- rc.sysstat serve per gestire il demone sysstat, che include strumenti per monitorare le prestazioni del sistema nel tempo, come: utilizzo della CPU, carico di I/O, attività dei dischi, traffico di rete, memoria, e altro.
- rc.sysvinit script di avvio utilizzato per eseguire compiti fondamentali di inizializzazione del sistema durante il processo di boot. È una parte importante del classico sistema di init (SysV-style) che Slackware continua a utilizzare. rc.sysvinit si occupa principalmente di: Impostare il runlevel corretto, Avviare servizi e script in /etc/rc.d/ in base al runlevel attivo, Gestire la transizione tra i runlevel (es. da single user a multi-user), Fare il "ponte" tra la struttura classica di Slackware e la compatibilità con alcuni script SysV-style, che potresti trovare in software non pensato espressamente per Slackware (come pacchetti di terze parti o provenienti da altre distro).
- rc.udev serve per avviare e gestire udev, il componente del sistema responsabile della gestione dinamica dei dispositivi (device management) nel sistema Linux. udev è parte di systemd, ma può funzionare anche in modalità standalone (come in Slackware). È il sistema che: Crea i device file in /dev (es: /dev/sda, /dev/ttyUSB0, ecc.) Gestisce dispositivi plug-and-play come USB, dischi, schede audio, webcam, ecc. Applica regole per nominare dispositivi, assegnare permessi, avviare script al collegamento di un device.

rc.ulogd script di init (startup script) utilizzato per avviare, fermare o gestire il servizio ulogd, ovvero User-space Logging Daemon per il sistema di logging dei pacchetti di rete. ulogd è un demone di logging usato insieme a iptables (e più recentemente anche a nftables) per registrare in modo dettagliato il traffico di rete. Diversamente dal logging standard tramite dmesg o syslog, ulogd permette un logging avanzato in formato strutturato, con possibilità di scrivere su: file di testo, database (MySQL, PostgreSQL, SQLite), JSON, syslog, ecc.

rc.wireless script di configurazione automatica per le interfacce di rete wireless. Non è uno script di init a sé stante (come rc.ulogd), ma viene chiamato automaticamente dallo script /etc/rc.d/rc.inet1 quando quest'ultimo gestisce un'interfaccia Wi-Fi.

rc.wireless.conf file di configurazione opzionale usato per fornire impostazioni avanzate e specifiche per le reti wireless, in particolare quando vengono rilevati determinati SSID.

rc.yppassd script di init usato per avviare, fermare o gestire i servizi NIS (Network Information Service), noti anche come Yellow Pages (YP).

rc0.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc1.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc2.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc3.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc4.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc5.d/	mantenuta per compatibilita'	col sistema di boot sysv
rc6.d/	mantenuta per compatibilita'	col sistema di boot sysv