

UNIX Security

UNIX is an operating system and also a philosophy of how an operating system should be. Linux is a fresh implementation of the UNIX philosophy implemented through the efforts of volunteers from across the globe. Linux runs on a multitude of hardware platforms.

UNIX Security Areas

- Log on security
- Password Storage
- File system security
- Process Security
- Directory Services
- Network Security
- System Logging and Auditing
- Security Protocol Support

Log on Security

- UNIX requires a valid user name and password to allow interactive log on
- This is true whether the log on is on the console or over the network
- The account must be present in the `/etc/passwd` file
- `/etc/issue` file – presents a `WARNING` message before the login prompt is issued
- `/etc/securetty` file – lists the tty's from which the root is allowed to log in
- `/etc/nologin` file – temporary disables interactive logins and prints the message
- `/etc/motd` – presents a welcome message after the login has been successful

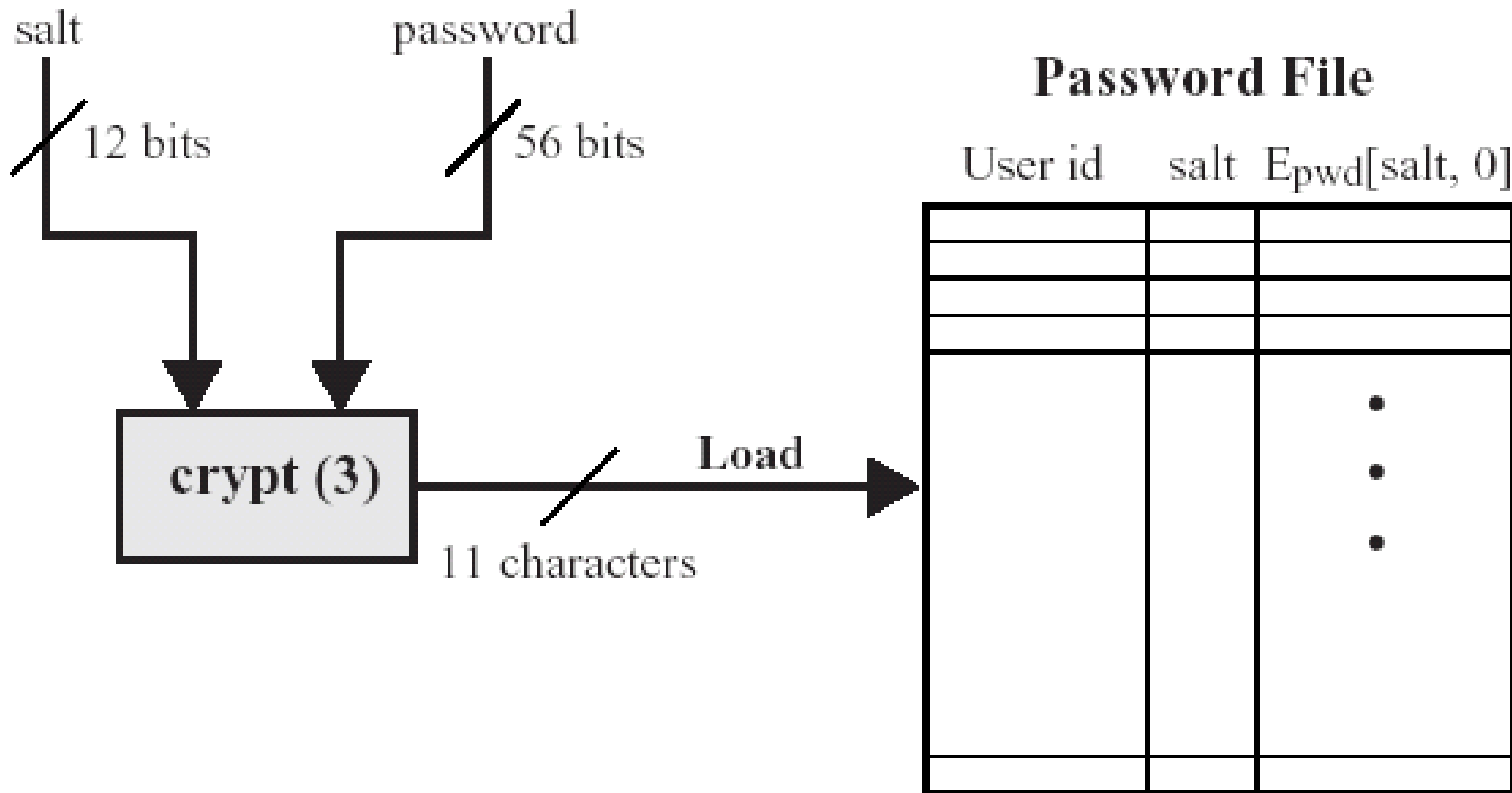
Password Storage

- Traditionally the account information is stored in the `/etc/passwd` file
- The `/etc/passwd` file is world-readable
- The encrypted version of the password is stored and not the password itself
- Other fields in a `/etc/passwd` entry – login name, user name, user id, group id, group, home directory, login shell,
- The SHADOW password system
- The MD5 password storage

/etc/passwd file format

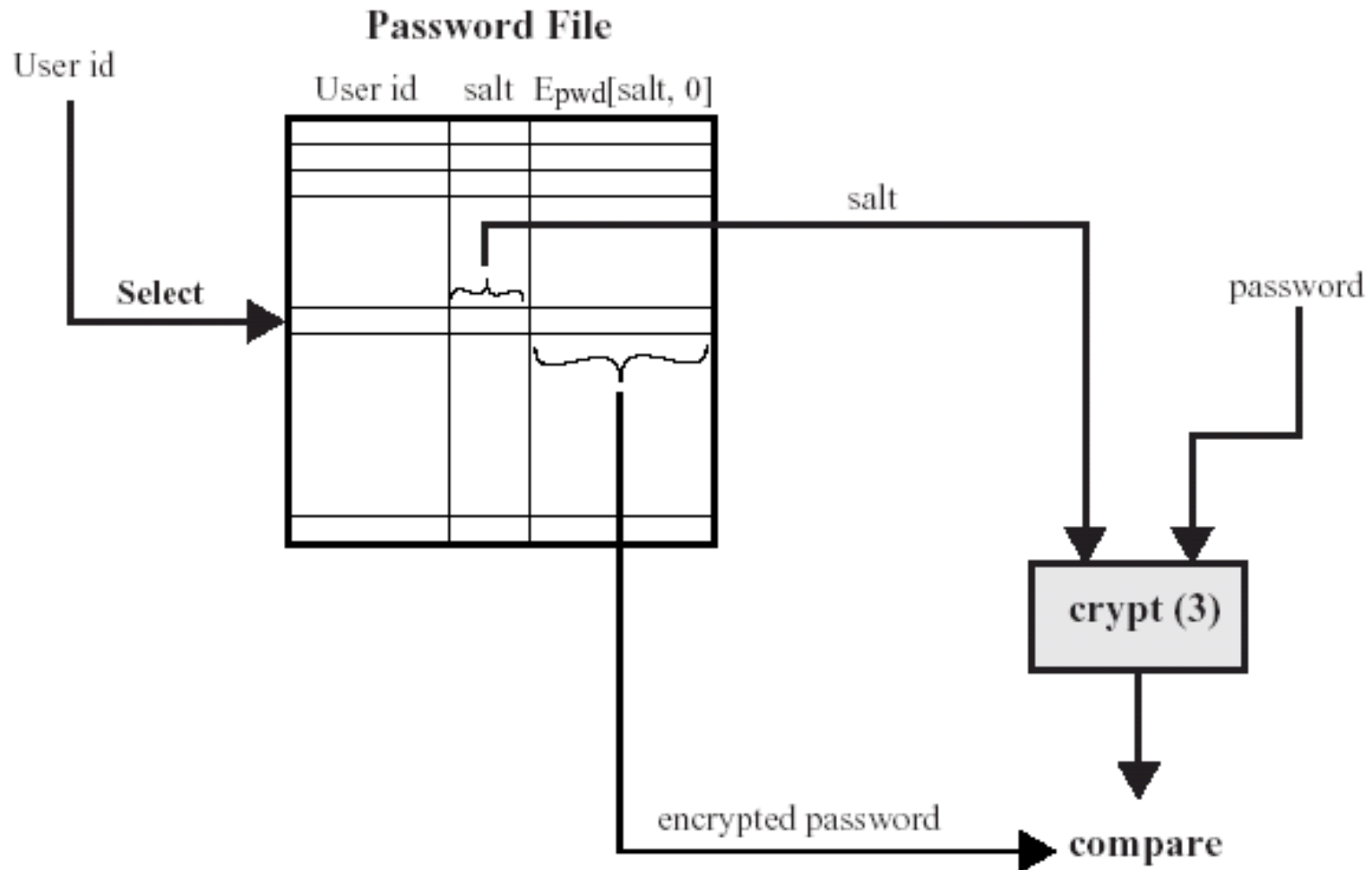
```
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:
operator:x:11:0:operator:/root:
games:x:12:100:games:/usr/games:
gopher:x:13:30:gopher:/usr/lib/gopher-data:
ftp:x:14:50:FTP User:/home/ftp:
nobody:x:99:99:Nobody:/:
xfs:x:43:43:X Font Server:/etc/X11/fs:/bin/false
named:x:25:25:Named:/var/named:/bin/false
piranha:x:60:60:./home/httpd/html/piranha:/dev/null
pvm:x:24:24:./usr/share/pvm3:/bin/bash
majordomo::91:91:Majordomo List Manager:/usr/lib/majordomo:/bin/bash
testing:x:53042:100:600:/home/testing:/usr/bin/pine
```

UNIX Password Scheme



(a) Loading a new password

UNIX Password Scheme



(b) Verifying a password

Shadow Passwords

- Shadow password system stores passwords in the file `/etc/shadow` which is not world-readable
- Password creation and update policies can be defined

MD5 Password Hash

- Stores MD5 hash of the password rather than encrypted password
- Hash functions are one-way and not-invertible
- That makes them more secure than plain encryption for this purpose

Rules to make passwords effective:

- They should be at least six characters in length, preferably eight characters including at least one numeral or special character.
- They must not be trivial; a trivial password is one that is easy to guess and is usually based on the user's name, family, occupation or some other personal characteristic.
- They should have an aging period, requiring a new password to be chosen within a specific time frame.
- They should be revoked and reset after a limited number of concurrent incorrect retries.

/etc/login.defs

```
C:\WINDOWS\System32\telnet.exe
IW /etc/login.defs (Read only) Row 11 Col 1 5:33 Ctrl-K H for help
#
# PASS_MAX_DAYS Maximum number of days a password may be used.
# PASS_MIN_DAYS Minimum number of days allowed between password changes
# PASS_MIN_LEN Minimum acceptable password length.
# PASS_WARN_AGE Number of days warning given before a password expires.
#
PASS_MAX_DAYS 99999
PASS_MIN_DAYS 0
PASS_MIN_LEN 5
PASS_WARN_AGE 7
#
# Min/max values for automatic uid selection in useradd
#
UID_MIN 500
UID_MAX 60000
#
# Min/max values for automatic gid selection in groupadd
#
GID_MIN 500
GID_MAX 60000
#
# If defined, this command is run when removing a user.
# It should remove any at/cron/print jobs etc. owned by
# the user to be removed (passed as the first argument).
#
#USERDEL_CMD /usr/sbin/userdel_local
```

File System Security

- UNIX supports file system level security
- File ownership
- File permissions
- chmod command
- Access Control List vs. File Permissions
- Support for filesystem level encryption is available through add-ons

Process Security

- Preemptive multi-tasking
- Memory Protection
- Programs run under user ID's so they are not allowed to access other user's data

Directory Services

- NIS
- NIS+
- LDAP
- Integration with Windows NT SAM

Pluggable Authentication Modules (PAM)

- PAM'ified applications can use any technology for authentication and security as required
- The Linux-PAM library allows the system administrator to choose how applications authenticate users, such as for console access, program and file access

Network Information Service

- Directory Service to centralize user accounts and group information
- Allows users with NIS Domain accounts to use any computer in the domain
- NIS uses the client server model
- Each NIS domain has a Master NIS server which has a master copy of the directory database
- Directory database is generated from flat-files and is in some binary format

Network Security

- The inetd super server
- TCP Wrappers
- Kernel level Firewall

The inetd Super-Server

- When running, inetd listens for connections on certain internet sockets.
- When a connection is found on one of its sockets, it looks up what service the socket corresponds to, and invokes a program to service the request.
- After the program is finished, INETD will continue to listen on the socket.

Benefits of INETD

- Essentially, inetd allows running one daemon to invoke several others, reducing load on the system.
- You can disable all services you do not want your system to offer by commenting them out
- If you change `/etc/inetd.conf` remember to 'killall -HUP inetd'

/etc/inetd.conf

```
C:\WINDOWS\System32\telnet.exe
#
# inetd.conf      This file describes the services that will be available
#                 through the INETD TCP/IP super server.  To re-configure
#                 the running INETD process, edit this file, then send the
#                 INETD process a SIGHUP signal.
#
# Version:       @(<#>)/etc/inetd.conf      3.10      05/27/93
#
# Authors:       Original taken from BSD UNIX 4.3/TAHOE.
#                 Fred N. van Kempen, <waltje@uwalt.nl.mugnet.org>
#
# Modified for Debian Linux by Ian A. Murdock <imurdock@oldell.portal.com>
#
# Modified for RHS Linux by Marc Ewing <marc@redhat.com>
#
# <service_name> <sock_type> <proto> <flags> <user> <server_path> <args>
#
# Echo, discard, daytime, and chargen are used primarily for testing.
#
# To re-read this file after changes, just do a 'killall -HUP inetd'
#
#echo    stream  tcp      nowait  root    internal
#echo    dgram  udp      wait    root    internal
#discard stream  tcp      nowait  root    internal
/etc/inetd.conf
```

TCP_WRAPPERS

- By default Red Hat Linux allows all service requests.
- Using TCP_WRAPPERS makes securing your servers against outside intrusion is a lot simpler and painless then you would expect.
- Deny all hosts by putting “ALL: ALL@ALL, PARANOID” in the “/etc/hosts.deny” file and explicitly list trusted hosts who are allowed to your machine in the “/etc/hosts.allow” file.
- This is the safest and the best configuration.

Configuring TCP_WRAPPERS

- TCP_WRAPPERS is controlled from two files and the search stops at the first match.
- /etc/hosts.allow
- /etc/hosts.deny
- Access will be granted when a (daemon, client) pair matches an entry in the /etc/hosts.allow file.
- Otherwise, access will be denied when a (daemon, client) pair matches an entry in the /etc/hosts.deny file
- Otherwise, access will be granted.

/etc/hosts.allow and /etc/hosts.deny

```
C:\WINDOWS\System32\telnet.exe
```

```
[athar@sirsyed athar]$ cat /etc/hosts.allow
```

```
##  
## hosts.allow This file describes the names of the hosts which are  
## allowed to use the local INET services, as decided  
## by the '/usr/sbin/tcpd' server.  
##
```

```
[athar@sirsyed athar]$ _
```

```
##  
## hosts.deny This file describes the names of the hosts which are  
## *not* allowed to use the local INET services, as decided  
## by the '/usr/sbin/tcpd' server.  
##
```

```
## The portmap line is redundant, but it is left to remind you that  
## the new secure portmap uses hosts.deny and hosts.allow. In particular  
## you should know that NFS uses portmap!
```

```
[athar@sirsyed athar]$
```

Kernel Level Firewall

- Linux has a built-in kernel level packet filter and circuit level gateway firewall
- You may use ipchains utility to manipulate the firewall rules
- Newer implementation is based on iptables and is a more powerful and feature-rich firewall
- Further information is available in Linux Firewall HOWTO

System Logging and Auditing

- Strong logging architecture
- syslogd logging daemon
- Applications also perform extensive logging
- Log files generally stored in /var/log/*
- Use `tail -f logfile` to see logging activity

Security Protocol Support

- All major network security protocols are supported:
- IPSec (<http://www.freeswan.org>)
- PPTP
- SSL-TLS (<http://www.openssl.org>)
- Kerberos
- PGP
- etc.