TESTBED Linux

SekChek for UNIX Security Report

System: Linuxwhite

9 November 2013

SekChek IPS
inbox@sekchek.com
www.sekchek.com

Declaration

The provided observations and recommendations are in response to a benchmarking analysis that compares the client's information security features against industry.

The recommendations are organised to identify possible implications to the company based on the gathered information, to identify an industry average rating of the controls and provide possible recommended actions.

The benchmarking analysis and the related observations and recommendations should supplement management's analysis but should not be and cannot be solely relied upon in any instance to identify and/or remediate information security deficiencies.

Further, the observations and recommendations herein do not identify the cause of a possible deficiency or the cause of any previously unidentified deficiencies. The causes of the deficiencies must be determined and addressed by management for the recommendations selected to be relevant.

Contents

SekChek Options	
Reference Number	1009090003
Requester	Richard Burns
Telephone Number	+44 (881) 846 8971
City	London
Client Country	UK
Charge Code	SekChek100909
Client Code	SEK001
Client Industry Type	Communications
Host Country	South Africa
Security Standards Template	0 - SekChek Default
Evaluate Against Industry Type	Communications
Compare Against Previous Analysis	Not Selected
Report Format	Word 2007
Paper Size	A4 (21 x 29.7 cms)
Spelling	English UK
Large Report Format	MS-Excel spreadsheet
Large Report (Max Lines in Word Tables)	200
Summary Document Requested	Yes
Scan Software Version Used	Version 5.1.0
Scan Software Release Date	08-Nov-2013

Your *SekChek* report was produced using the above options and parameters.

You can change these settings for all files you send to us for processing via the *Options* menu in the *SekChek* Client software on your PC. You can also tailor them (i.e. temporarily override your default options) for a specific file via the *Enter Client Details* screen. This screen is displayed:

- For *SekChek* for NT and NetWare - during the Extract process on the target Host system;
- For *SekChek* for AS/400 and UNIX - during the file encryption process in the *SekChek* Client software.

System Details	
Host Name	Linuxwhite
Scan Time	04-Nov-2013 07:58
Operating System	Linux
OS Release	2.4.18-3
OS Version	#1 Thu Apr 18 07:37:53 EDT 2002
Machine	i686
Scanned By	root

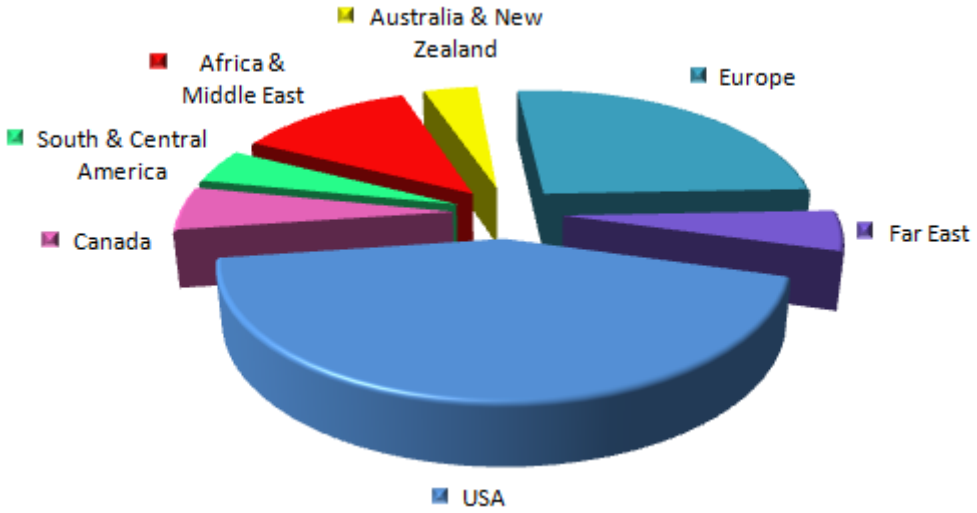
Report Date: 9 November, 2013

1 . Report Summaries

The following two charts illustrate the diversity of regions and industries that make up the population of *UNIX systems* in our statistics database. The remaining graphs in the *Report Summary* section evaluate security on your system against this broad base of real-life security averages.

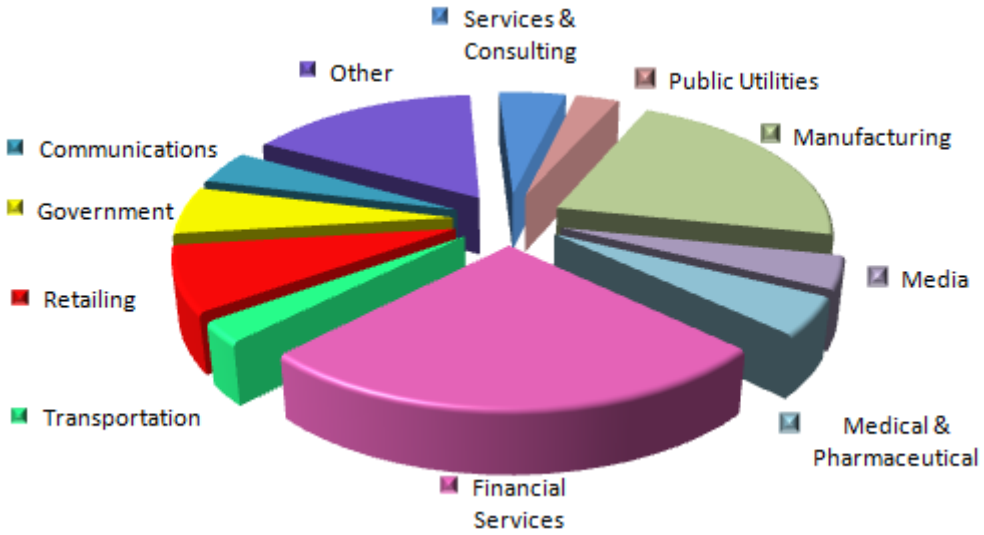
SekChek is used by the Big Four audit firms, IS professionals, internal auditors, security consultants & general management in more than 130 countries.

Statistics Population by Region



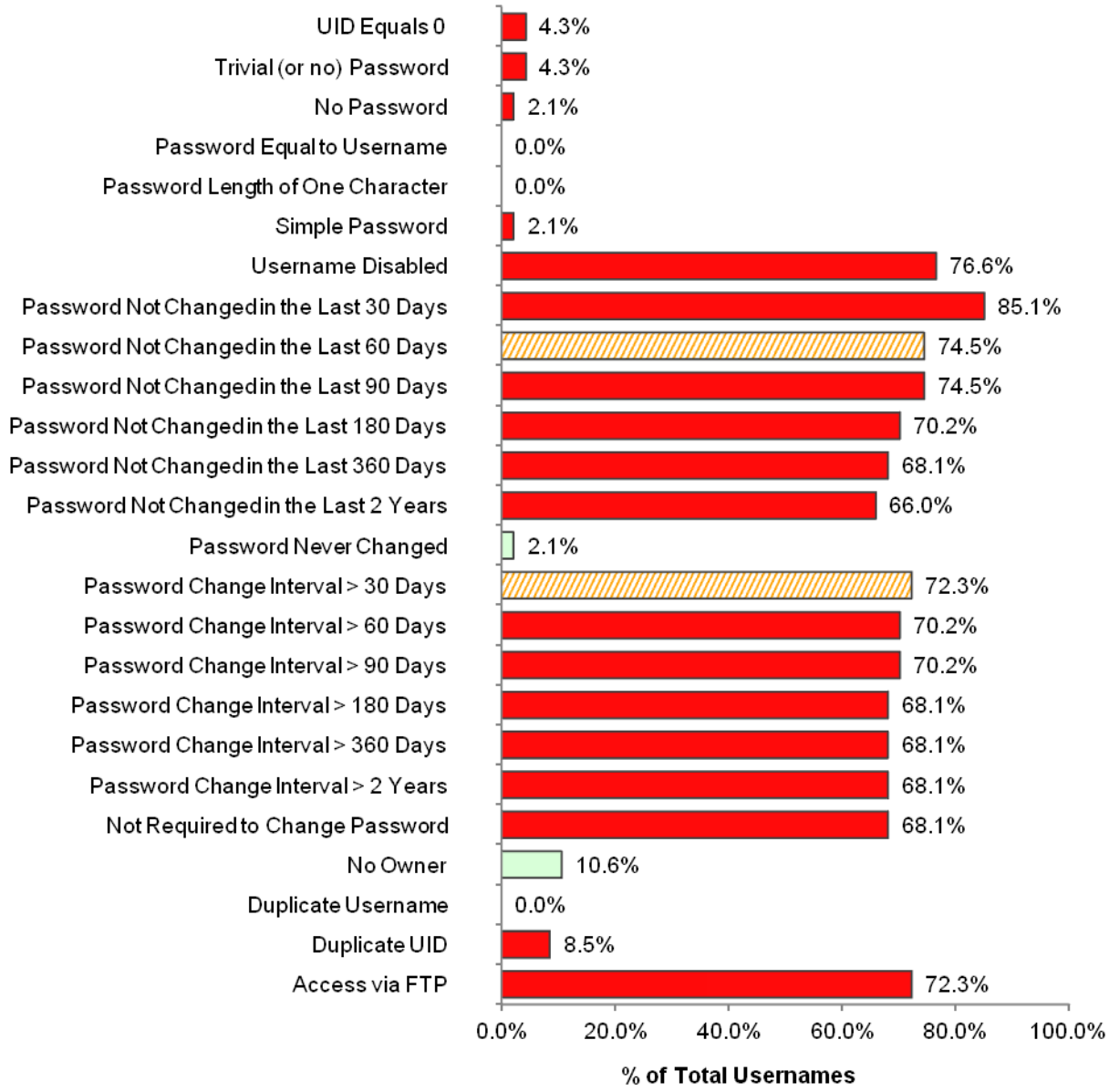
As new reviews are processed, summaries of the results (excluding client identification) are automatically added to a unique statistics database containing more than 70,000 assessments.

Statistics Population by Industry Type



1.1 Comparisons Against Industry Average and Leading Practice

Summary of Usernames



This graph compares against the industry average using the following criteria:
Country = <All>; Industry Type = Communications; Machine Size (Nbr of Usernames) = Medium

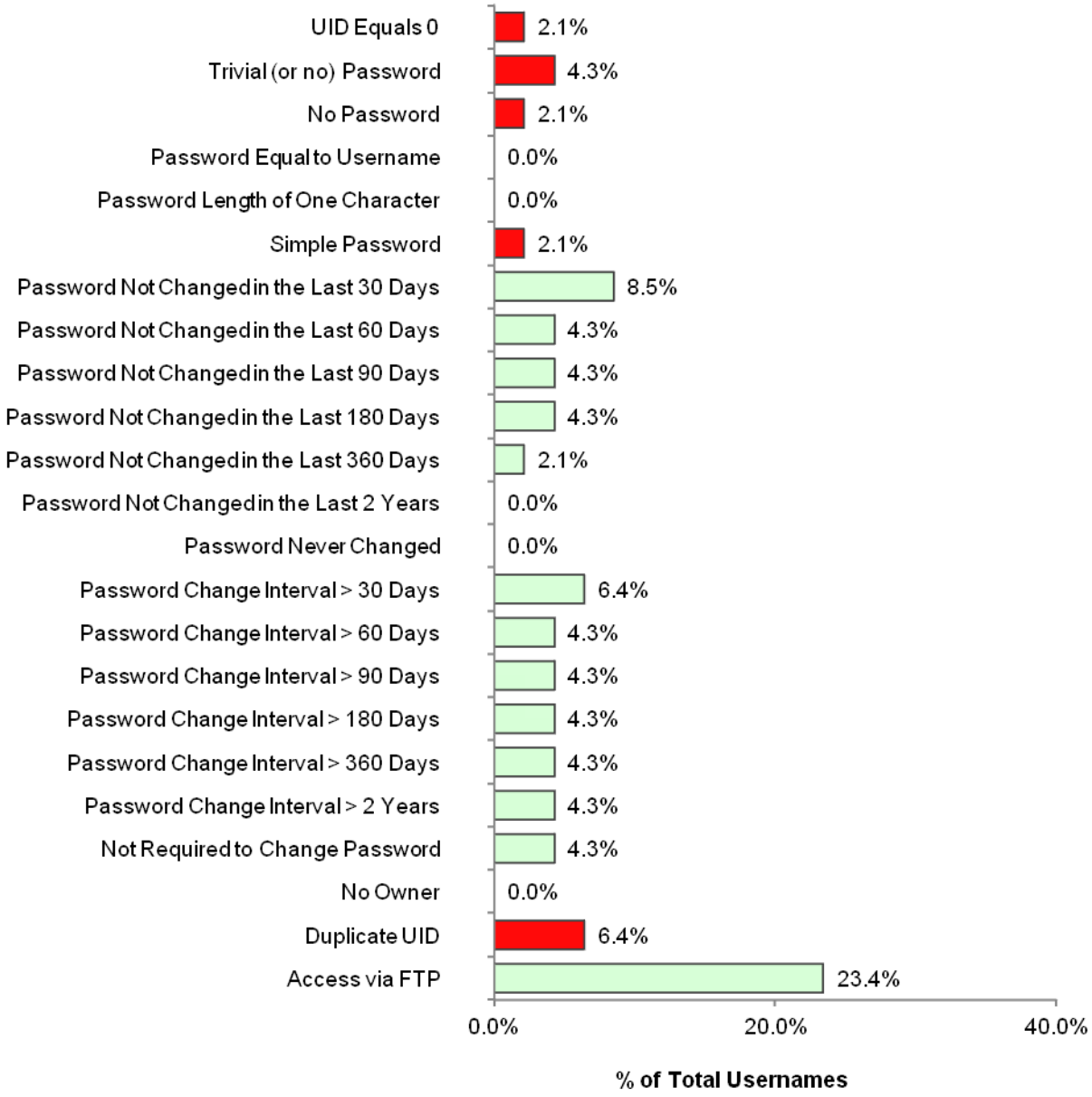
Above the industry average; About average; Below average

Total number of usernames defined to your system: 47.

This summary report presents the number of usernames, with the listed characteristics, as a percentage of the total number of usernames defined to your system. In general, longer bars highlight potential weaknesses in your security measures and should be investigated. For more details, refer to the relevant section in the ***main body*** of the report.

Comparisons Against Industry Average and Leading Practice (continued)

Summary of Usernames (excluding disabled usernames)



This graph compares against the industry average using the following criteria:
Country = <All>; Industry Type = Communications; Machine Size (Nbr of Usernames) = Medium

Above the industry average; About average; Below average

Total number of usernames defined to your system: 47.

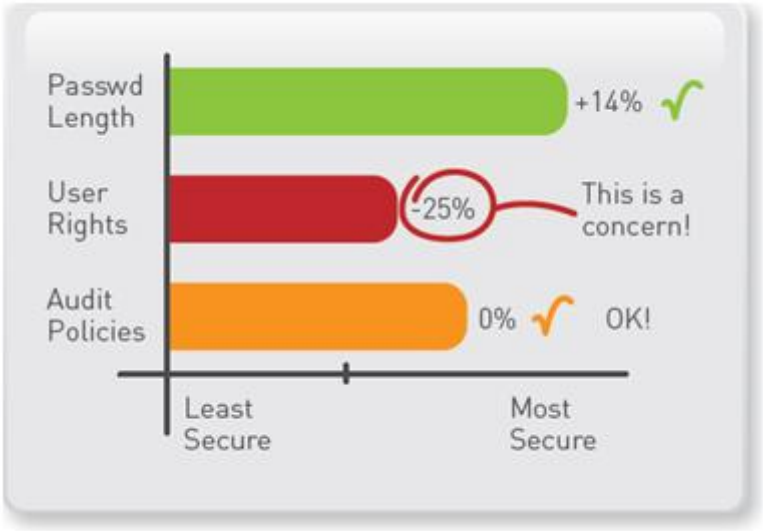
This summary report presents the number of *enabled* usernames (i.e. excluding usernames with disabled passwords) with the listed characteristics, as a percentage of the total number of usernames defined to your system.

In general, longer bars highlight potential weaknesses in your security measures and should be investigated. For more details, refer to the relevant sections in the main body of the report.

1.2 Summary of Changes since the Previous Analysis

Need to quickly highlight changes in security controls since your previous review?

SekChek's latest time-comparison graphs are just the solution!



Note: The above graph is provided for illustrative purposes only.

A collection of easy-to-read reports in a very familiar format provides you with visual indicators of:

- Whether security has improved, weakened, or remained about the same since your previous analysis
- The effectiveness of your measures to strengthen controls
- Whether risk is increasing or decreasing
- The degree of change, both positive and negative

The applications are endless. Some of the practical benefits are:

- Time savings. Reduced time spent poring over volumes of unconnected information
- Objectivity. The results are guaranteed to be the same regardless of who performs the review
- Compliance with legislation. Easier monitoring for compliance with statutory requirements imposed by SOX, HIPAA and other legislative changes relating to corporate governance
- More powerful justifications. The ability to present more convincing arguments to senior, non-technical management who do not have the time, or the inclination, to understand masses of technical detail

Interested?

Contact us at **inbox@sekchek.com** to find out how to get started!

2 . System-Wide Security Policy

This report section lists the host's system-wide security policy settings and compares them with *leading practice* values.

Policy Value	Current Value	Leading Practice
Minimum Password Length	10	8 or greater
Minimum Password Change Interval	5	0
Maximum Password Change Interval	30	60 days or less
Password Expiry Warning (days)	3	1 or greater

Explanation of the Policy Values

Policy Value	Description
Minimum Password Length	The minimum acceptable password length. The risk of unauthorised access to your system increases if password lengths are too short. You will also lose accountability for actions performed by usernames.
Minimum Password Change Interval	The minimum number of days allowed between password changes. A value of '0' (no restriction) is recommended so that users can change their passwords *immediately* if they suspect they are known by another person. However, this setting can increase the risk of passwords remaining the same, despite system-enforced changes. This is because users could change their passwords several times in quick succession until they are set back to the original value.
Maximum Password Change Interval	The maximum number of days that a password may be used. If set too high, a successful intruder is effectively permitted to use a compromised account for a longer period of time, before the account owner changes the password. Long periods between password changes also increase the risk of passwords becoming common knowledge.
Password Expiry Warning (days)	A password expiry warning is issued this number of days before the password expires.

Notes:

- Settings for *Minimum Password Change Interval, Maximum Password Change Interval* and *Password Expiry Warning* are only used at the time of account creation. Any changes to these settings will not affect existing accounts.
- The *Minimum Password Length* parameter defined in file /etc/login.defs typically has no affect if the pam_cracklib module is used. This is because the strength of user passwords is controlled by the pam_cracklib module.

3 . Password Shadowing

Section Summary

The system **IS** using password shadowing features.

Implications

Systems *without* password shadowing features store users' passwords in encrypted format in the system's */etc/passwd* file, which is typically world-readable. This makes user passwords very vulnerable to decryption by software-driven attempts to crack passwords.

Password shadowing features reduce this risk by storing users' encrypted passwords in a shadow file, such as */etc/shadow*, which can only be read by privileged accounts, such as *root*.

Risk Rating

High. (if password shadowing features are not installed and enabled)

Recommended Action

If password shadowing features are not currently installed and enabled on your system you should seriously consider implementing these features.

4 . Usernames, UIDs and Home Directories

Section Summary

There are a total of 47 usernames defined to your system:

- 4.3% (2) of usernames have a UID of 0 (equivalent to root)
- 10.6% (5) of usernames are not assigned to an Owner

Section Detail

Username	UID	Owner Name	Home Directory	Shell
adm	3	adm	/var/adm	/sbin/nologin
bin	1	bin	/bin	/sbin/nologin
daemon	2	daemon	/sbin	/sbin/nologin
disableduser	507	Test account (Vassie)	/home/disableduser	/bin/bash
enableduser	506	Test account (Vassie)	/home/enableduser	/bin/bash
ftp	14	FTP User	/var/ftp	/sbin/nologin
games	12	games	/usr/games	/sbin/nologin
gdm	42		/var/gdm	/sbin/nologin
gopher	13	gopher	/var/gopher	/sbin/nologin
halt	7	halt	/sbin	/sbin/halt
ident	98	pident user	/	/sbin/nologin
kevin	500	Kevin Tromp	/home/kevin	/bin/bash
lp	4	lp	/var/spool/lpd	/sbin/nologin
mail	8	mail	/var/spool/mail	/sbin/nologin
mailnull	47		/var/spool/mqueue	/dev/null
mandla	508	Mandla Ncube	/home/mandla	/bin/bash
named	25	Named	/var/named	/bin/false
news	9	news	/var/spool/news	
nfsnobody	65534	Anonymous NFS User	/var/lib/nfs	/sbin/nologin
ninon	513	Ninon Nkulu	/home/ninon	/bin/bash
nobody	99	Nobody	/	/sbin/nologin
nopwd	514	nopwd_user (Ninon)	/home/nopwd	/bin/bash
nscd	28	NSCD Daemon	/	/bin/false
ntp	38		/etc/ntp	/sbin/nologin
operator	11	operator	/root	/sbin/nologin
pcap	77		/var/arpwatch	/sbin/nologin
radvd	75	radvd user	/	/bin/false
root	0	Root,,,	/root	/bin/bash
rpc	32	Portmapper RPC user	/	/sbin/nologin
rpcuser	29	RPC Service User	/var/lib/nfs	/sbin/nologin
rpm	37		/var/lib/rpm	/bin/bash
sarah	501	Sarah Singh	/home/sarah	/bin/bash
sektest	502	Test account (Ninon)	/home/sektest	/bin/bash
shutdown	6	shutdown	/sbin	/sbin/shutdown
sync	5	sync	/sbin	/bin/sync
test01	504	Test account (Vassie)	/home/test01	/bin/bash

Username	UID	Owner Name	Home Directory	Shell
test02	505	Test account (Vassie)	/home/test02	/bin/bash
test11	510	Test for software release (Mandla)	/home/test11	/bin/bash
testgroupuser	503	Test account (Vassie)	/home/testgroupuser	/bin/bash
up+	512	Test account (Ninon)	/home/up+	/bin/bash
up=	511	Test account (Ninon)	/home/up=	/bin/bash
user01	509	Test for SSH	/home/user01	/bin/bash
uucp	10	uucp	/var/spool/uucp	/sbin/nologin
vassie	508	Vassie Pather	/home/vassie	/bin/bash
vcsa	69	virtual console memory owner	/dev	/sbin/nologin
xfs	43	X Font Server	/etc/X11/fs	/bin/false
yasir	0	Yasir Butt	/home/yasir	/bin/bash

Implications

In general, usernames should be assigned to specific individuals and owners should be responsible for ensuring the confidentiality of their private login passwords. If usernames are assigned to job functions, and shared by several people, it will be difficult to ensure accountability for actions performed by them.

Usernames that are no longer in use, such as those belonging to personnel who have since left the organisation, should be promptly deleted from the system. Redundant usernames present intruders with unnecessary opportunities to gain access to your system with little risk of detection.

Risk Rating

Medium to High. (If usernames are not assigned to specific individuals)

Recommended Action

You should check that:

- Usernames are still current and that their owners still require access to the system;
- The number of usernames with a UID of 0 is not excessive;
- Usernames are assigned to specific individuals and not to job functions; and
- User's home directories and shell environments are appropriate.

You should also ensure that the password for the root account is known by a maximum of two or three people only.

5 . Groups and their Members

Section Summary

There are a total of 60 groups, containing the following members, defined on your system:

- 23.3% (14) of the groups do not contain any members and might be redundant (see 'Analysis of Redundant Groups')

This report details the members of the various groups defined on your system. The 'Src' field indicates whether the username is defined as a group member in the system's */etc/passwd* ('P') or */etc/group* ('G') files, or in both.

Where group membership is gained via an entry in the */etc/passwd* file ('P'), this is also the user's *primary* group. Each user can have only *one* primary group.

Where group membership is gained via entries in the */etc/group* file ('G'), these are the user's *secondary* groups. Users can belong to *many* secondary groups.

Section Detail

Group Name	GID	Group Members	Src	Owner Name
adm	4	adm	P	adm
		adm	G	adm
		daemon	G	daemon
		root	G	Root,,,
bin	1	bin	P	bin
		bin	G	bin
		daemon	G	daemon
		root	G	Root,,,
daemon	2	bin	G	bin
		daemon	P	daemon
		daemon	G	daemon
		root	G	Root,,,
disableduser	508	disableduser	P	Test account (Vassie)
disk	6	root	G	Root,,,
enableduser	507	enableduser	P	Test account (Vassie)
ftp	50	ftp	P	FTP User
		halt	G	halt
gdm	42	gdm	P	
gopher	30	gopher	P	gopher
ident	98	ident	P	pident user
kelly2	516	mandla	P	Mandla Ncube
lp	7	daemon	G	daemon
		lp	P	lp
		lp	G	lp
mail	12	mail	P	mail
		mail	G	mail
mailnull	47	mailnull	P	
named	25	named	P	Named
news	13	news	P	news
		news	G	news
nfsnobody	65534	nfsnobody	P	Anonymous NFS User

Group Name	GID	Group Members	Src	Owner Name
ninonk	513	ninon	P	Ninon Nkulu
nobody	99	nobody	P	Nobody
nopwd	518	nopwd	P	nopwd_user (Ninon)
nscd	28	nscd	P	NSCD Daemon
ntp	38	ntp	P	
pcap	77	pcap	P	
radvd	75	radvd	P	radvd user
root	0	halt	P	halt
		operator	P	operator
		root	P	Root,,,
		root	G	Root,,,
		shutdown	P	shutdown
		sync	P	sync
rpc	32	rpc	P	Portmapper RPC user
rpcuser	29	rpcuser	P	RPC Service User
rpm	37	rpm	P	
sekchek_users	500	kevin	P	Kevin Tromp
		sarah	P	Sarah Singh
sektest	502	sektest	P	Test account (Ninon)
sys	3	adm	G	adm
		bin	G	bin
		root	G	Root,,,
test01	505	test01	P	Test account (Vassie)
test02	506	test02	P	Test account (Vassie)
test11	510	test11	P	Test for software release (Mandla)
testgroupuser	504	testgroupuser	P	Test account (Vassie)
up+	512	up+	P	Test account (Ninon)
up=	511	up=	P	Test account (Ninon)
user01	501	user01	P	Test for SSH
users	100	games	P	games
uucp	14	uucp	P	uucp
		uucp	G	uucp
vassien	509	vassie	P	Vassie Pather
vcsa	69	vcsa	P	virtual console memory owner
wheel	10	root	G	Root,,,
xfs	43	xfs	P	X Font Server
YasirB	517	yasir	P	Yasir Butt

Implications

Group profiles are a convenient way to provide multiple users with the same set of access permissions and privileges. Access permissions assigned to group profiles are added to permissions that are directly assigned to Users via their usernames.

If users are assigned to groups with excessive permissions to system resources, they will have access to unnecessary system functions and information resources, which could be abused and used to exploit security on your system.

Risk Rating

Medium to High. (If users are assigned to groups with excessive permissions)

Recommended Action

You should review the above listing to ensure that usernames (group members) are assigned to the correct groups.

Where a User is defined as a group member in the PASSWD file ('Src' = 'P') ***and*** in the GROUP file ('Src' = 'G'), you should consider removing one of the two entries as a matter of good housekeeping.

6 . Discrepancies in Passwd And Shadow Passwd Files

Section Summary

Your system's /etc/passwd and shadow passwd files contain 2 discrepancies:

- 1 usernames exist in your system's /etc/passwd file, but not in the shadow passwd file
- 1 usernames exist in your system's shadow passwd file, but not in the /etc/passwd file

Section Detail

Username	Comment
bill	In Shadow PASSWD file only
yasir	In PASSWD file only

Implications

This report section highlights discrepancies between your system's normal (/etc/passwd) and shadow *passwd* files. It lists:

- Accounts defined in your *normal* passwd file that are not defined in your shadow passwd file; and
- Accounts defined in your *shadow* passwd file that are not defined in your normal passwd file.

Such discrepancies often occur when *passwd* files are maintained with standard editing software, rather than with the software vendor's maintenance utilities.

This report most likely indicates a housekeeping issue.

Risk Rating

None. (A housekeeping issue only)

Recommended Action

You should ensure the appropriate vendor-supplied utilities are always used to maintain your system's *passwd* files. These utilities help maintain the integrity of your system's *passwd* and shadow *passwd* files and keep them in synchronisation.

You should ensure that any unintended entries are removed from these files.

7 . Duplicate Usernames, UIDs & GIDs

Section Summary

0.0% (0) of the usernames defined to your system are duplicates.
8.5% (4) of the UIDs (User Identifiers) defined to your system are duplicates.
3.3% (2) of the GIDs (Group Identifiers) defined to your system are duplicates.

Section Detail

Username	UID	GID	Group Name	Comment
		100	defgroup	Duplicate GID
		100	users	Duplicate GID
mandla	508			Duplicate UID
root	0			Duplicate UID
vassie	508			Duplicate UID
yasir	0			Duplicate UID

Implications

UID = User Identifier; GID = Group Identifier.

This report highlights duplicate entries in your system's *primary* PASSWD and GROUP files (i.e. not the Shadow PASSWD and GROUP files, if these are used on your system). Duplicate entries often occur when the files are updated using a general edit program, rather than a specially designed PASSWD editor.

Although the report mainly highlights issues of a housekeeping nature, certain situations can have a significant impact on the security of your system.

For example, UNIX identifies users and groups by their numeric UID and GID, rather than their usernames and group names. The implication here is that if two users are assigned the same UID, the system 'sees' them as the same user, even though they have different usernames and passwords. I.e. their security privileges and permissions would be identical.

The security implications for group names and GIDs are similar to those for usernames and UIDs.

Risk Rating

Medium to High.

Recommended Action

Unless it is intentional, you should ensure that all users and groups are assigned their own unique UIDs and GIDs.

You should also ensure that duplicate usernames are removed from your PASSWD file.

8 . Password Change Intervals

Section Summary

All Accounts

72.3% (34) of users are not required to change their passwords every 30 days (or less):

- 70.2% (33) of users are not required to change their passwords every 60 days (or less)
- 70.2% (33) of users are not required to change their passwords every 90 days (or less)
- 68.1% (32) of users are not required to change their passwords every 180 days (or less)
- 68.1% (32) of users are not required to change their passwords every year (or less)
- 68.1% (32) of users are not required to change their passwords every 2 years (or less)
- 68.1% (32) of users are never required to change their passwords

Excluding Disabled Accounts

6.4% (3) of users are not required to change their passwords every 30 days (or less):

- 4.3% (2) of users are not required to change their passwords every 60 days (or less)
- 4.3% (2) of users are not required to change their passwords every 90 days (or less)
- 4.3% (2) of users are not required to change their passwords every 180 days (or less)
- 4.3% (2) of users are not required to change their passwords every year (or less)
- 4.3% (2) of users are not required to change their passwords every 2 years (or less)
- 4.3% (2) of users are never required to change their passwords

Industry Average Comparison (> 30 days)

Section Detail

Maximum Password Change Interval	Username	UID	Owner Name	Disabled
99999	adm	3	adm	Yes
99999	bin	1	bin	Yes
99999	daemon	2	daemon	Yes
99999	ftp	14	FTP User	Yes
99999	games	12	games	Yes
99999	gdm	42		Yes
99999	gopher	13	gopher	Yes
99999	halt	7	halt	Yes
99999	ident	98	pident user	Yes
99999	lp	4	lp	Yes
99999	mail	8	mail	Yes
99999	mailnull	47		Yes
99999	named	25	Named	Yes
99999	news	9	news	Yes
99999	nfsnobody	65534	Anonymous NFS User	Yes
99999	ninon	513	Ninon Nkulu	
99999	nobody	99	Nobody	Yes
99999	nopwd	514	nopwd_user (Ninon)	
99999	nscd	28	NSCD Daemon	Yes
99999	ntp	38		Yes
99999	operator	11	operator	Yes
99999	pcap	77		Yes
99999	radvd	75	radvd user	Yes

Maximum Password Change Interval	Username	UID	Owner Name	Disabled
99999	rpc	32	Portmapper RPC user	Yes
99999	rpcuser	29	RPC Service User	Yes
99999	rpm	37		Yes
99999	shutdown	6	shutdown	Yes
99999	sync	5	sync	Yes
99999	uucp	10	uucp	Yes
99999	vcsa	69	virtual console memory owner	Yes
99999	xfs	43	X Font Server	Yes
99999	yasir	0	Yasir Butt	Yes
111	test01	504	Test account (Vassie)	Yes
45	root	0	Root,,,	
30	disableduser	507	Test account (Vassie)	Yes
30	enableduser	506	Test account (Vassie)	Yes
30	kevin	500	Kevin Tromp	
30	mandla	508	Mandla Ncube	
30	sarah	501	Sarah Singh	
30	sektest	502	Test account (Ninon)	
30	test02	505	Test account (Vassie)	
30	test11	510	Test for software release (Mandla)	
30	testgroupuser	503	Test account (Vassie)	Yes
30	up+	512	Test account (Ninon)	Yes
30	up=	511	Test account (Ninon)	Yes
30	user01	509	Test for SSH	
30	vassie	508	Vassie Pather	

Implications

Password Change Intervals are expressed in days. A value of '99999' indicates that regular password changes are not enforced. Note that some of these accounts may be disabled.

Passwords that are not changed on a frequent basis are vulnerable to being compromised over time. This would enable an intruder to gain unauthorised access to your system functions and data.

Risk Rating

Medium to High. (Dependant on the strength of other password controls, such as minimum password length and rules governing password content)

Recommended Action

Password change intervals for these user accounts should be brought in line with installation standards.

A generally accepted standard is to force users to change their passwords every 30 to 60 days.

9 . Redundant Groups & Group Members

Section Summary

23.3% (14) of the groups defined on your system do not contain any members.
1 usernames are defined as group members, but do not exist in your PASSWD file.

Section Detail

Group Name	GID	Group Member	Comment
defgroup	100	sanjayp	Redundant Group Member
dip	40		Redundant Group
floppy	19		Redundant Group
games	20		Redundant Group
grouptest	503		Redundant Group
kelly	515		Redundant Group
kmem	9		Redundant Group
lock	54		Redundant Group
man	15		Redundant Group
mem	8		Redundant Group
slocate	21		Redundant Group
superuser	520		Redundant Group
thierryk	514		Redundant Group
tty	5		Redundant Group
utmp	22		Redundant Group

Implications

A housekeeping issue only.

Risk Rating

None.

Recommended Action

You should determine the reason these redundant groups and group members are defined to your system. If there is no valid reason, they should be deleted from your system.

10 . Disabled Usernames

Section Summary

76.6% (36) of usernames are disabled and cannot be used to sign-on to your system.

Industry Average Comparison

Section Detail

Username	UID	Owner Name
adm	3	adm
bin	1	bin
daemon	2	daemon
disableduser	507	Test account (Vassie)
enableduser	506	Test account (Vassie)
ftp	14	FTP User
games	12	games
gdm	42	
gopher	13	gopher
halt	7	halt
ident	98	pident user
lp	4	lp
mail	8	mail
mailnull	47	
named	25	Named
news	9	news
nfsnobody	65534	Anonymous NFS User
nobody	99	Nobody
nscd	28	NSCD Daemon
ntp	38	
operator	11	operator
pcap	77	
radvd	75	radvd user
rpc	32	Portmapper RPC user
rpcuser	29	RPC Service User
rpm	37	
shutdown	6	shutdown
sync	5	sync
test01	504	Test account (Vassie)
testgroupuser	503	Test account (Vassie)
up+	512	Test account (Ninon)
up=	511	Test account (Ninon)
uucp	10	uucp
vcsa	69	virtual console memory owner
xfs	43	X Font Server

Username	UID	Owner Name
yasir	0	Yasir Butt

Implications

A housekeeping issue only.

Risk Rating

None.

Recommended Action

You should determine the reason why these usernames are disabled. If they are inactive and no longer required, they should be deleted from your system.

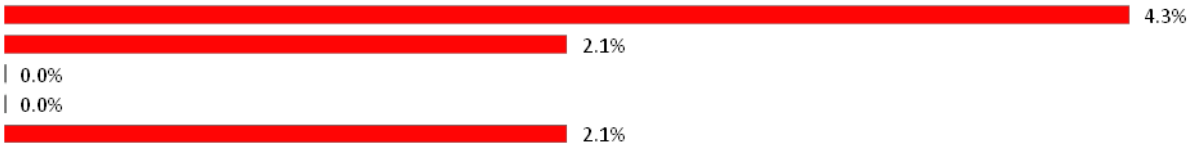
11 . Trivial Passwords

Section Summary

4.3% (2) of usernames defined to your system have a trivial (or no) password:

- 2.1% (1) of usernames do not have a password
- 0.0% (0) of usernames have a password equal to the username
- 0.0% (0) of usernames have a password that is only one character long
- 2.1% (1) of usernames have a simple password

Industry Average Comparison



Note. SekChek's password assessment routines simulate a manual / casual password guessing attempt using the first 8 characters (only) of a user password. This emulates the functionality on older UNIX systems, which only consider the first 8 characters of a password to be significant. For example, if account 'bettysue' has a password of 'bettysue1', SekChek will indicate that the password is trivial and equal to the user name.

SekChek highlights user passwords that are: blank (zero-length); equal to the username; only 1 character in length; and commonly used words. Examples of commonly used words are: days of the week; months of the year; 'secret', 'password', 'qwerty' etc.

Section Detail

Username	UID	Owner Name	Trivial Password
nopwd	514	nopwd_user (Ninon)	No Password
test11	510	Test for software release (Mandla)	A simple word

Implications

Weak passwords, such as those assigned to the above usernames, increase the risk of unauthorised access to your system and information resources. The particular resources an intruder could gain access to depends on the file permissions and privileges assigned to the username.

Weak password controls also result in a loss of accountability for actions performed on your system.

Risk Rating

High.

Recommended Action

Unless there is a valid reason for not restricting access to these usernames, you should ensure strong passwords are assigned to *all* usernames defined to your system.

This is achieved through a combination of user education and discipline, and the implementation of software-enforced controls, such as minimum lengths and password content. The range of software-enforced controls available on your particular machine is largely dependent on the type and version of your Operating System.

12 . Passwords, 30 Days and Older

Section Summary

All Accounts
85.1% (40) of the usernames on your system have not had their passwords changed in the last month:
- 74.5% (35) of usernames have not had their passwords changed in the last 2 months
- 74.5% (35) of usernames have not had their passwords changed in the last 3 months
- 70.2% (33) of usernames have not had their passwords changed in the last 6 months
- 68.1% (32) of usernames have not had their passwords changed in the last year
- 66.0% (31) of usernames have not had their passwords changed in the last 2 years
- 2.1% (1) of usernames have never had their passwords changed

Excluding Disabled Accounts
8.5% (4) of the usernames on your system have not had their passwords changed in the last month:
- 4.3% (2) of usernames have not had their passwords changed in the last 2 months
- 4.3% (2) of usernames have not had their passwords changed in the last 3 months
- 4.3% (2) of usernames have not had their passwords changed in the last 6 months
- 2.1% (1) of usernames have not had their passwords changed in the last year
- 0.0% (0) of usernames have not had their passwords changed in the last 2 years
- 0.0% (0) of usernames have never had their passwords changed

The password for the root account was last changed 1 days ago.

Industry Average Comparison (> 30 days)

Note. This is an exception report, so only lists profiles whose passwords have not changed in the last 30 days. I.e. if a profile's password was changed 29 days ago (or more recently) it will not be listed in the report section.

Section Detail

Password Last Changed	Username	UID	Owner Name	Password Change Interval	Disabled
	yasir	0	Yasir Butt	99999	Yes
23-Jul-2008	adm	3	adm	99999	Yes
23-Jul-2008	bin	1	bin	99999	Yes
23-Jul-2008	daemon	2	daemon	99999	Yes
23-Jul-2008	ftp	14	FTP User	99999	Yes
23-Jul-2008	games	12	games	99999	Yes
23-Jul-2008	gdm	42		99999	Yes
23-Jul-2008	gopher	13	gopher	99999	Yes
23-Jul-2008	halt	7	halt	99999	Yes
23-Jul-2008	ident	98	pident user	99999	Yes
23-Jul-2008	lp	4	lp	99999	Yes
23-Jul-2008	mail	8	mail	99999	Yes
23-Jul-2008	mailnull	47		99999	Yes
23-Jul-2008	named	25	Named	99999	Yes
23-Jul-2008	news	9	news	99999	Yes
23-Jul-2008	nfsnobody	65534	Anonymous NFS User	99999	Yes
23-Jul-2008	nobody	99	Nobody	99999	Yes
23-Jul-2008	nscd	28	NSCD Daemon	99999	Yes
23-Jul-2008	ntp	38		99999	Yes

Password Last Changed	Username	UID	Owner Name	Password Change Interval	Disabled
23-Jul-2008	operator	11	operator	99999	Yes
23-Jul-2008	pcap	77		99999	Yes
23-Jul-2008	radvd	75	radvd user	99999	Yes
23-Jul-2008	rpc	32	Portmapper RPC user	99999	Yes
23-Jul-2008	rpcuser	29	RPC Service User	99999	Yes
23-Jul-2008	rpm	37		99999	Yes
23-Jul-2008	shutdown	6	shutdown	99999	Yes
23-Jul-2008	sync	5	sync	99999	Yes
23-Jul-2008	uucp	10	uucp	99999	Yes
23-Jul-2008	vcsa	69	virtual console memory owner	99999	Yes
23-Jul-2008	xfs	43	X Font Server	99999	Yes
13-Sep-2009	disableduser	507	Test account (Vassie)	30	Yes
27-Oct-2012	ninon	513	Ninon Nkulu	99999	
26-Dec-2012	sarah	501	Sarah Singh	30	
18-May-2013	up+	512	Test account (Ninon)	30	Yes
18-May-2013	up=	511	Test account (Ninon)	30	Yes
22-Sep-2013	test01	504	Test account (Vassie)	111	Yes
22-Sep-2013	test02	505	Test account (Vassie)	30	
26-Sep-2013	enableduser	506	Test account (Vassie)	30	Yes
04-Oct-2013	test11	510	Test for software release (Mandla)	30	
04-Oct-2013	testgroupuser	503	Test account (Vassie)	30	Yes

Implications

Note. HP-UX systems that are not trusted systems (i.e. that do not use HP's C2-level trusted mode) only store the week in which a user password was last changed - a 'week' starts on a Thursday and ends on a Wednesday. This means that it is not possible to report the precise date of the last password change because it could have occurred anytime between a Thursday and Wednesday. On these systems the value reported in the Password Last Changed column is the last day of the 'week'. This is always a Wednesday.

Values of 100% probably indicate that controls over password ageing are not used on your system (in which case *SekChek* will report that login passwords have never been changed for all users), or that they are set to inappropriate levels.

Inappropriate password change intervals increase the risks of unauthorised access to your system, and passwords becoming common knowledge amongst system users.

Risk Rating

Medium to High. (Dependent on the strength of password controls, such as minimum password length and password content)

Recommended Action

These usernames should be reviewed and deleted if they are redundant and no longer required. Otherwise, their password change intervals should be brought in-line with installation standards.

A generally accepted standard is to force users to change their passwords every 30-60 days.

13 . Last Logins

Section Summary

Note that this report is based on information obtained from your system's ‘wtmp’ file, so does not report on any logins prior to the earliest recorded entry in this file.

Section Detail

Last Logon	Username	Owner Name
	adm	adm
	bin	bin
	daemon	daemon
	ftp	FTP User
	games	games
	gdm	
	gopher	gopher
	halt	halt
	ident	pident user
	lp	lp
	mail	mail
	mailnull	
	named	Named
	news	news
	nfsnobody	Anonymous NFS User
	nobody	Nobody
	nopwd	nopwd_user (Ninon)
	nscd	NSCD Daemon
	ntp	
	operator	operator
	pcap	
	radvd	radvd user
	rpc	Portmapper RPC user
	rpcuser	RPC Service User
	rpm	
	shutdown	shutdown
	sync	sync
	uucp	uucp
	vcsa	virtual console memory owner
	xfs	X Font Server
01-Apr-2013	disableduser	Test account (Vassie)
09-Sep-2013	user01	Test for SSH
15-Sep-2013	vassie	Vassie Pather
19-Oct-2013	enableduser	Test account (Vassie)
19-Oct-2013	up+	Test account (Ninon)
19-Oct-2013	up=	Test account (Ninon)

Last Logon	Username	Owner Name
20-Oct-2013	sektest	Test account (Ninon)
26-Oct-2013	test01	Test account (Vassie)
26-Oct-2013	test11	Test for software release (Mandla)
26-Oct-2013	testgroupuser	Test account (Vassie)
29-Oct-2013	test02	Test account (Vassie)
03-Nov-2013	ninon	Ninon Nkulu
03-Nov-2013	yasir	Yasir Butt
04-Nov-2013	kevin	Kevin Tromp
04-Nov-2013	mandla	Mandla Ncube
04-Nov-2013	root	Root,,,
04-Nov-2013	sarah	Sarah Singh

Implications

Inactive and redundant usernames provide intruders with unnecessary opportunities to gain access to your system with little risk of detection.

Risk Rating

Low to Medium. (If redundant usernames exist in your security file)

Recommended Action

You should check the above list of usernames to ensure that there are no inactive and redundant usernames defined to your system. Redundant usernames should be deleted or disabled.

14 . System Search Path

Section Detail

Directory
/usr/local/sbin
/sbin
/usr/sbin
/bin
/usr/bin
/usr/bin/X11
/usr/local/bin
/usr/bin
/usr/X11R6/bin
/root/bin
/root/bin

Implications

The system searches the directories in the *System Search Path* to determine a program's location when it is requested to execute a program where a directory name is not specified.

If the *directories* in the system search path, or the *files* in these directories, are not properly secured it is possible to compromise security by:

- Adding extra directories to the system search path. This allows an intruder to insert his own version of a common program (e.g. *login*) into these directories.
- Changing the sequence of directories in the search path. If different programs with the same name appear in more than one of these directories, an incorrect version of the program may be executed.
- Substituting a common program (e.g. *login*) with another with a different function. An example is a special version of the *login* program that writes users' passwords to a hidden file.

Note that it is possible to amend the list of directories in the system search path via the system login script and individual user login scripts.

Risk Rating

High. (If tampered with)

Recommended Action

You should ensure that all directories in the system search path are necessary and justified. You should also ensure that write permissions (in particular) to these directories, as well as the files in the directories, are adequately restricted.

See report sections ***Permissions on selected Sensitive Directories*** and ***Permissions on selected Sensitive Files*** and for details of permissions.

15 . System Login Script File

Section Summary

Your system-wide login script file contains the following commands, which are executed at login time for all system users.

Section Detail

System Login Script
pathmunge () {
if ! echo $PATH | /bin/egrep -q "(^|:)$1($|:)" ; then
if ["$2" = "after"] ; then
PATH=$PATH:$1
else
PATH=$1:$PATH
fi
fi
if [`id -u` = 0]; then
pathmunge /sbin
pathmunge /usr/sbin
pathmunge /usr/local/sbin
fi
pathmunge /usr/X11R6/bin after
unset pathmunge
ulimit -S -c 0 > /dev/null 2>&1
USER="`id -un`"
LOGNAME=$USER
MAIL="/var/spool/mail/$USER"
HOSTNAME=`/bin/hostname`
HISTSIZE=1000
if [-z "$INPUTRC" -a ! -f "$HOME/.inputrc"]; then
INPUTRC=/etc/inputrc
fi
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE INPUTRC
for i in /etc/profile.d/*.sh ; do
if [-r "$i"]; then
. $i
fi
done
unset i

Implications

The system login script file (/etc/profile or /etc/login) is used to ensure that certain programs and commands are executed for every user that logs on to the system. It is executed before a user's personal login script (usually called .profile or .login) executes.

By inserting commands into system or user login scripts it is possible for a hacker to undermine security and cause serious damage to your system. Examples include commands that delete files or replace original files and programs with modified copies or impostors.

Risk Rating

High. (If tampered with)

Recommended Action

You should periodically check the contents of the system login script to ensure that it has not been tampered with and contains authorised commands only.

You should also check the file's permissions list to ensure that unintended people do not have write access to it.

Ensure any umask definitions for users in /etc/profile or ~/.profile files are set to: 0077 (-rwx------) - best; 0037 (-rwxr-----); or 0027 (-rwxr-x---).

16 . Files with World-Writeable Permissions

Section Summary

Your system contains 13 world-writeable files and directories.

Section Detail

For details see worksheet ***World_Writeable_Files*** in the MS-Excel workbook.

Implications

Every user defined on your system has write access to the above list of files and directories. Even if this is your intention, you should consider the impact of unauthorised changes or deletions to these objects.

Consider also the possible impact of a file being run by root (or under the root account), which has been defined or modified by an intruder.

Another consideration is the potential for 'denial of service' attacks as a result of large amounts of meaningless data being dumped on your system by a mischievous intruder.

Note that access permissions defined for a sub-directory override those permissions defined at higher levels in the directory tree structure.

Risk Rating

High. (If world-writeable access is granted to sensitive files)

Recommended Action

You should check the list of files and directories and ensure they are intended to be world-writeable.

You should keep the number of world-writeable files and directories to the minimum.

You should also endeavour to ensure that anything run by root is:

- Owned by root;
- NOT Group or World-writeable;
- In a directory where every directory in the path is owned by root and is not Group- or World-writeable.

17 . Permissions on selected Sensitive Files

Section Summary

This report details the permissions for 233 of the more sensitive files on your system:
- 0.4% (1) of these have group-writeable permissions
- 0.0% (0) of these have world-writeable permissions

Notes for values in the 'Other Info' column:
- *'HD' indicates that the file resides in a user's home directory;*
- *'SP' indicates that the file resides in a directory in the system Search Path.*

Section Detail

For details see worksheet ***Sensitive_Files*** in the MS-Excel workbook.

Implications

This report section lists access permissions on:

- Sensitive system configuration files (e.g. */etc/passwd*, */etc/inetd.conf*);
- Sensitive files in users' home directories (e.g. *.rhosts* files);
- *All* files and programs in *all* directories in the System search path (the system searches this path, to determine a program's location, each time it is requested to execute a program).

Some of these files contain statements that are 'executed' by the system when certain services are started and initialised. These commands can impact on the security of your system.

These files are obvious targets for people who might wish to substitute their own programs or insert other parameters and commands, to circumvent security or cause damage to your system.

Risk Rating

High. (If access permissions are inappropriate and allow unintended write access)

Recommended Action

You should periodically check the contents of these files to ensure they have not been tampered with and that unauthorised statements have not been inserted.

You should also ensure that permissions are reasonable and that ownership is correctly assigned.

Examples of recommended permissions for some of these files are:

/etc/default/login	-rw-r—r—	(644)	/etc/passwd	-rw-r—r--	(644)
/etc/exports	-rw-r—r—	(644)	/etc/services	-rw-r—r--	(644)
/etc/hosts.equiv	-rw-------	(600)	/etc/shadow	-r--------	(400)
/etc/hosts.lpd	-rw-------	(600)	/etc/ttys	-rw-r—r--	(644)
/etc/inetd.conf	-rw-------	(600)	~/.rhosts	-rw-------	(600)

18 . Permissions on selected Sensitive Directories

Section Summary

This section lists 257 of the more sensitive directories on your system.

Section Detail

For details see worksheet ***Sensitive_Directories*** in the MS-Excel workbook.

Implications

Sensitive directories, such as those containing files that determine the operating environment and security rules for a service, are prime targets of hackers. Hackers often attempt to insert their own files in these directories, in order to circumvent security or cause damage to a system.

Note that access permissions defined for a sub-directory override those permissions defined at higher levels in the directory tree structure.

Risk Rating

High. (If these directories allow write access to unintended usernames).

Recommended Action

You should ensure that permission lists for these directories are appropriate and do not allow write access to unintended users.

19 . SUID Permissions

Section Summary

Your system contains 30 programs with the SUID (Switch User Identification) permission set:

- 0.0% (0) of these have group-writeable permissions
- 0.0% (0) of these have world-writeable permissions

Section Detail

Program Name	Owner	Group	Permissions	Group/World Writeable
/bin/mount	root	root	-rwsr-xr-x	
/bin/ping	root	root	-rwsr-xr-x	
/bin/su	root	root	-rwsr-xr-x	
/bin/umount	root	root	-rwsr-xr-x	
/sbin/pwdb_chkpwd	root	root	-r-sr-xr-x	
/sbin/unix_chkpwd	root	root	-r-sr-xr-x	
/usr/bin/at	root	root	-rwsr-xr-x	
/usr/bin/chage	root	root	-rwsr-xr-x	
/usr/bin/chfn	root	root	-rws--x--x	
/usr/bin/chsh	root	root	-rws--x--x	
/usr/bin/crontab	root	root	-rwsr-xr-x	
/usr/bin/gpasswd	root	root	-rwsr-xr-x	
/usr/bin/kcheckpass	root	root	-rwsr-xr-x	
/usr/bin/lppasswd	root	root	-rwsr-xr-x	
/usr/bin/newgrp	root	root	-rws--x--x	
/usr/bin/passwd	root	root	-r-s--x--x	
/usr/bin/rcp	root	root	-rwsr-xr-x	
/usr/bin/rlogin	root	root	-rwsr-xr-x	
/usr/bin/rsh	root	root	-rwsr-xr-x	
/usr/bin/ssh	root	root	-rwsr-xr-x	
/usr/bin/sudo	root	root	---s--x--x	
/usr/lib/mc/bin/cons.saver	vcsa	root	-rws--x--x	
/usr/sbin/ping6	root	root	-rwsr-xr-x	
/usr/sbin/sendmail.sendmail	root	root	-r-sr-xr-x	
/usr/sbin/traceroute	root	root	-rwsr-xr-x	
/usr/sbin/traceroute6	root	root	-rwsr-xr-x	
/usr/sbin/userhelper	root	root	-rws--x--x	
/usr/sbin/userisdnctl	root	root	-rwsr-xr-x	
/usr/sbin/usernetctl	root	root	-rwsr-xr-x	
/usr/X11R6/bin/XFree86	root	root	-rws--x--x	

Implications

Programs with SUID permission assume the UID of their Owner (typically ‘root’) when they execute. Many system utilities and commands, such as back-ups, mail and password change programs, execute with SUID ‘root’.

If a user executing one of these programs manages to 'break out' of the program, the person can assume ‘root’ privileges and gain *unrestricted* access to your system and information. One aspect to consider is whether the

program really needs to execute under the powerful root account. Perhaps it could perform the same function under a username or group with fewer permissions.

If an SUID program gives unintended *write* access to users, your system is *very* exposed. This is because someone could replace the SUID program with a program that has a different function to the original and use it to gain access to 'root' at any time.

For example, it would be easy to copy a UNIX shell (e.g. /bin/sh) over the SUID program. A hacker could then gain access to 'root' at any time by simply executing this 'special version' of the SUID program.

Refer also the report ***SGID Permissions***.

Risk Rating

High. (If incorrectly specified)

Recommended Action

The above list of programs should be checked to ensure that they are legitimate programs that require the powerful SUID privilege.

You should also check that:

- Unauthorised changes have not been made to any of these programs;
- The programs are being executed from the intended directories;
- The associated Owner is appropriate and is not too powerful (i.e. does not have excessive permissions) for the program's function; and
- The program's permission list does not allow write access to users who do not require it.

20 . SGID Permissions

Section Summary

Your system contains 21 programs with the SGID (Switch Group Identification) permission set:

- 0.0% (0) of these have group-writeable permissions
- 0.0% (0) of these have world-writeable permissions

Section Detail

Program Name	Owner	Group	Permissions	Group/World Writeable
/sbin/netreport	root	root	-rwxr-sr-x	
/usr/bin/gataxx	root	games	-r-xr-s--x	
/usr/bin/glines	root	games	-r-xr-s--x	
/usr/bin/gnibbles	root	games	-r-xr-s--x	
/usr/bin/gnobots2	root	games	-r-xr-s--x	
/usr/bin/gnome-stones	root	games	-r-xr-s--x	
/usr/bin/gnomine	root	games	-r-xr-s--x	
/usr/bin/gnotravex	root	games	-r-xr-s--x	
/usr/bin/gnotski	root	games	-r-xr-s--x	
/usr/bin/gtali	root	games	-r-xr-s--x	
/usr/bin/iagno	root	games	-r-xr-s--x	
/usr/bin/kdesud	root	root	-rwxr-sr-x	
/usr/bin/lockfile	root	mail	-rwxr-sr-x	
/usr/bin/mahjongg	root	games	-r-xr-s--x	
/usr/bin/same-gnome	root	games	-r-xr-s--x	
/usr/bin/slocate	root	slocate	-rwxr-sr-x	
/usr/bin/wall	root	tty	-r-xr-sr-x	
/usr/bin/write	root	tty	-rwxr-sr-x	
/usr/sbin/gnome-pty-helper	root	utmp	-rwxr-sr-x	
/usr/sbin/lockdev	root	lock	-rwxr-sr-x	
/usr/sbin/utempter	root	utmp	-rwxr-sr-x	

Implications

Programs with SGID permission assume the GID of their group when they execute. Many system utilities and commands, such as ‘passwd’, ‘lp’, ‘pstat’ and ‘ps’ use this.

However, if a user executing one of these programs manages to 'break out' of the program, the person can assume the privileges and permissions of the program's group and use this to exploit security on your system.

Another risk is that someone with write permission to any of these programs could replace the original program with his own and gain access to the privileges of the group under which the program executes.

Note that on most UNIX systems, an upper-case ‘S’ in the group-writeable field indicates that the file has SGID permission only, whereas a lower-case ‘s’ indicates that the file has the SGID permission set AND is group-executable (i.e. it can be executed by all members of the file’s group).

Refer also the report ***SUID Permissions***.

Risk Rating

High. (If incorrectly specified)

Recommended Action

The above list of programs should be checked to ensure they are legitimate programs that require the SGID privilege.

You should also check that:

- Unauthorised changes have not been made to any of these programs;
- The programs are being executed from the intended directories;
- The associated group is appropriate and does not have too many permissions for the program's function; and
- The program's permission list does not allow write access to users who do not require it.

21 . Network Services

Section Summary

The xinetd daemon manages 22 network services on your system.

- 2 of these services are enabled, 20 are disabled.

Section Detail

Service	Socket	Protocol	Disabled	User	Server Path	Arguments
chargen	dgram	udp	yes	root		
chargen	stream	tcp	yes	root		
daytime	dgram	udp	yes	root		
daytime	stream	tcp	yes	root		
defaults						
echo	dgram	udp	yes	root		
echo	stream	tcp	yes	root		
exec	stream		yes	root	/usr/sbin/in.rexecd	
finger	stream		yes	nobody	/usr/sbin/in.fingerd	
ftp	stream		no	root	/usr/sbin/in.ftpd	-l -a
login	stream		yes	root	/usr/sbin/in.rlogind	
ntalk	dgram		yes	nobody	/usr/sbin/in.ntalkd	
ntalk	dgram		yes	root	/usr/bin/ktalkd	
rsync	stream		yes	root	/usr/bin/rsync	--daemon
servers	stream	tcp	yes			
services	stream	tcp	yes			
sgi_fam	stream	tcp		root	/usr/bin/fam	
shell	stream		yes	root	/usr/sbin/in.rshd	
talk	dgram		yes	nobody	/usr/sbin/in.talkd	
talk	dgram		yes	root	/usr/bin/kotalkd	
telnet	stream		yes	root	/usr/sbin/in.telnetd	
time	dgram	udp	yes	root		
time	stream	tcp	yes	root		

The listed network services are started automatically by the system's inetd (or xinetd) daemon. Note that:

- Some of these services may not have been active at the time *SekChek* was run. For example, services may have been halted by the system administrator.
- Additional services may have been started by the system administrator.

Service.
The identifying label of the service invoked by the *inetd* daemon. Some of the more common services are:

- *chargen*. Returns copies of the printable subset of ASCII characters. Used for network diagnostics.
- *defaults*. This is not a service. It contains the default settings for services managed by xinetd.
- *finger*. Returns information about users on the host machine.
- *ftp* (file transfer protocol). Used for transferring files between host machines.
- *netstat*. Returns network status information to the remote system.
- *nfs* (network file system). Responsible for handling client requests for NFS file sharing.
- *rsh*. Enables the execution of commands on a remote system.
- *rstat*. Returns performance statistics about the host system.
- *rusers*. Returns information about users on the host machine.

- *telnet*. Provides user login services.
- *tftp* (trivial file transfer protocol). Mainly used to support the remote boot facility for diskless workstations and router devices.
- *uucp*. Responsible for the transfer of UUCP data over the network.

Socket Type.
The type of the data delivery service. Typically either *stream*, a byte-oriented service provided by TCP; *dgram*, a transaction-oriented service provided by UDP; or *raw*, which runs directly on IP.

Protocol.
The name of the transport protocol. Typically *tcp* or *udp*.

Disabled.
Indicates whether the service is enabled (*no* or blank) or disabled (*yes*).

Wait.
Indicates that *inetd* must wait for the service protocol to release the socket connecting it to the network before *inet* can resume listening for more requests on that socket. *Nowait* enables *inetd* to immediately listen for more requests on the socket.

Server Path.
The full path name of the program that *inetd* must invoke. A Server Path of *internal* is typically used for small and non-demanding servers, which are implemented as part of the *inetd* server itself.

Arguments.
Command line arguments passed to the server service.

Implications

Network services expose your system in 3 ways:

- Some services such as *ftp* and *telnet*, provide a *direct* access path to your system from external equipment;
- Other services such as *finger*, provide intruders with *information* about your system (e.g. details of inactive user accounts), which can be used to gain access to your system;
- Thirdly, every network service has known and unknown security loopholes and risks. Because many services run under *root*, any bug in the software can enable an intruder to gain access to your superuser account and therefore to all resources on your system.

Most vendors correct reported security flaws fairly promptly so in general, older versions of software contain more security holes than more recent versions.

Another risk is that, if an intruder can gain write access to file /etc/inetd.conf (or /etc/xinetd.conf) or to the server software itself, he could for example, change one of the above service entries to give himself a root shell at any time.

Risk Rating

Medium to High. (If inappropriate network services are made available)

Recommended Action

You should check the list of network services to ensure they are valid and required and that you have applied all available software 'patches', particularly those addressing known security exposures. *The best general advice is if you do not really need a service, rather disable it or remove it from your system.*

You can disable your system's ability to receive connections by removing, or commenting out, the relevant line in file /etc/inetd.conf.

Ensure that the file /etc/inetd.conf (or /etc/xinetd.conf) is owned by the root account and permissions are set to 0600 or 0644 (-rw------- or -rw-r--r--). See report ***Permissions on selected Sensitive Files***.

You should also check the software versions you are using and consider upgrading to the latest version available from your vendor.

22 . Current Network Connections

Section Summary

There are 17 active network connections on your system.

Section Detail

Protocol	Local Address	Local Port	Remote Address	Remote Port	State
tcp	0.0.0.0	111	0.0.0.0	*	LISTEN
tcp	0.0.0.0	139	0.0.0.0	*	LISTEN
tcp	0.0.0.0	21	0.0.0.0	*	LISTEN
tcp	0.0.0.0	22	0.0.0.0	*	LISTEN
tcp	0.0.0.0	32768	0.0.0.0	*	LISTEN
tcp	0.0.0.0	6000	0.0.0.0	*	LISTEN
tcp	127.0.0.1	25	0.0.0.0	*	LISTEN
tcp	127.0.0.1	32769	0.0.0.0	*	LISTEN
tcp	200.200.100.49	139	200.200.100.56	49927	ESTABLISHED
udp	0.0.0.0	137	0.0.0.0	*	
udp	0.0.0.0	138	0.0.0.0	*	
udp	0.0.0.0	32768	0.0.0.0	*	
udp	127.0.0.1	32769	0.0.0.0	*	
udp	200.200.100.106	137	0.0.0.0	*	
udp	200.200.100.106	138	0.0.0.0	*	
udp	200.200.100.49	137	0.0.0.0	*	
udp	200.200.100.49	138	0.0.0.0	*	

Protocol
The name of the transfer protocol. Typically *tcp, udp or raw.*

Local Address
The address of the local end of the socket.

Local Port
The port number of the local end of the socket.

Remote Address
The address of the remote end of the socket.

Remote Port
The port number of the remote end of the socket.

State
Shows the connection state of the socket. There are no states defined in *raw* mode and usually no states defined in *udp* and therefore, this column may be blank. This can be one of the following values:

BOUND	Bound, ready to connect or listen
CLOSE_WAIT	The remote end has shut down, waiting for the socket to close
CLOSED	The socket is not being used
CLOSING	Both sockets are shut down but we still don't have all our data sent
ESTABLISHED	The socket has an established connection
FIN_WAIT1	The socket is closed and the connection is shutting down
FIN_WAIT2	The connection is closed and the socket is waiting for a shutdown from the remote end
IDLE	Idle, opened but not bound
LAST_ACK	The remote end has shut down and the socket is closed. Waiting for acknowledgement
LISTEN	The socket is listening for incoming connections
SYN_RECV	A connection request has been received from the network
SYN_SENT	The socket is actively attempting to establish a connection

TIME_WAIT	The socket is waiting after close to handle packets still in the network
UNKNOWN	The state of the socket is unknown

Implications

This report section lists all active network connections for TCP and UDP protocols, including the local and remote addresses, the ports in use and the state of each connection. It does not indicate which services are configured to use these ports. Note that the information in this section will be less useful for UDP because it is a *connectionless* protocol.

The port numbers used by some of the most common network services are:

Port number	Service
20	ftp data
21	ftp
22	ssh
23	telnet
25	smtp
43	whois
53	DNS
69	tftp
79	finger
80	http
110	POP3
143	IMAP
161	snmp
443	https

Network services and their associated ports provide several opportunities for intruders to exploit your system. Some examples are:

- Services such as telnet (port 23) and ftp (port 21) transmit user passwords in clear text format, which makes them vulnerable to access via 'sniffer' software;
- Older versions of services often contain security weaknesses, which can be exploited to gain access to your system using the account under which the service is run (often 'root');
- Services such as finger (port 79), provide intruders with useful information about your system, such as details of inactive user accounts, which can be used to gain access to your system.

Risk Rating

Medium to High. (If inappropriate network services are running)

Recommended Action

You should determine what services are configured to use these ports and:

- Disable any unused or redundant services;
- Limit the number of services that run under the 'root' account by running them under an account with less privileges (e.g. *nobody*);
- Frequently check with your software vendor for security vulnerabilities in the services you are running and apply any relevant software patches;
- Consider replacing services that transmit passwords in clear text format with more secure software. E.g. use *ssh* (secure shell), rather than telnet;
- Ensure that hosts running open services are located behind properly configured firewall machines;
- Monitor open ports and connections for signs of unusual activity, particularly from addresses external to your organisation.

You should also ensure that write permissions on sensitive configuration files, such as /etc/inetd.conf and /etc/services, are restricted to the appropriate accounts.

23 . Trusted Hosts

Section Summary

There are 1 Trusted Hosts defined to your system.

Section Detail

Trusted Host	Username
pinkpanther	root
pinkpanther	test01

Implications

If a remote machine is defined to your system as trusted, any user with the same username on both hosts can login to your system, using the .rsh and .rlogin commands, without typing a password.

Although the trust concept offers many advantages, particularly for users that regularly access several host machines, the security implications are worth considering.

For example, unless security on a trusted host, over which you may have little control, is maintained at the same standard as your system, users on the trusted host can undermine the majority of your security efforts around your machine.

Inappropriately defined trusted hosts have been the cause of many of the security breaches reported in recent years.

Refer also the report ***Trusted Users***.

Risk Rating

Medium to High. (If incorrectly specified or if security on any trusted host is not maintained at the same level as your system)

Recommended Action

You should check the list of trusted hosts and ensure they are valid and appropriate. In general, you should only trust those hosts under your domain or management.

You should also satisfy yourself that the quality of security on trusted hosts is at a sufficiently high standard, so that it does not undermine security on your system.

Ensure that the first character of the /etc/hosts.equiv file is not '-', that there are no '!' or '#' characters in the file and that you do not have a '+' entry by itself. This may allow any outside user to gain access to your system.

Ensure that the permissions on your /etc/hosts.equiv file are 0600 (-rw-------) and that the file is owned by root. Refer to the report ***Permissions on selected Sensitive Files***.

If you are not running 'r' commands, you should have no use for the /etc/hosts.equiv file and it can probably be deleted.

24 . Trusted Users

Section Summary

There are 0 .rhosts files on your system:

- 0.0% (0) of these have group-writeable permissions
- 0.0% (0) of these have world-writeable permissions

Section Detail

*** None were found ***

Implications

Trusted users are similar to ***trusted hosts***, except they refer to individual users and not to entire hosts. If a user on another system is designated as a trusted user for your system, the user can login to your system, via the .rsh and .rlogin commands, without providing a login password.

In many respects there is more risk attached to the concept of trusted users than with trusted hosts.

This is due to the difficulties in controlling the definition of trusted users on your system; all a user need do is include the required statements in a *.rhosts* file in his home directory. Hackers often search for *.rhosts* files and attempt to insert a username of their choice in them.

For these reasons, many organisations choose to disallow or disable the concept of trusted users.

Risk Rating

High. (If permissions are inappropriate or *.rhosts* files are not properly controlled)

Recommended Action

You should regularly inspect the contents of the *.rhosts* files defined on your system and ensure they are valid and necessary.

Ensure that *.rhosts* files are owned by the account's owners and that permissions only allow write access by the owner.

Ensure that the first character in these files is not '-', that they do not contain '!' or '#' characters and that they do not contain '+' on any line. This may allow any user access to these accounts.

Unless the trusted user concept is absolutely essential you should consider disabling it by removing the feature from the source code, or ensuring *.rhosts* files are automatically deleted from your system. Review on a case-by-case basis.

25 . Users Not Allowed Access via FTP

Section Summary

27.7% (13) of usernames are prohibited from accessing your system via FTP.

Section Detail

Username	UID	Owner Name
adm	3	adm
bin	1	bin
daemon	2	daemon
games	12	games
halt	7	halt
lp	4	lp
mail	8	mail
news	9	news
nobody	99	Nobody
operator	11	operator
shutdown	6	shutdown
sync	5	sync
uucp	10	uucp

Implications

If powerful usernames are allowed to log in via FTP, your system resources and information are unnecessarily exposed to unauthorised access, tampering and damage. Hackers could, for example, carry out 'denial of service' attacks on your system by transferring very large files to your system, which could deny access to legitimate system users.

Many older versions of FTP server software contained bugs that enabled hackers to break into a system.

Like other network services, such as Telnet, passwords transmitted to FTP are sent 'in the clear' which raises the possibility of them being intercepted and logged by 'sniffer' software. This information could then be used to gain unauthorised access to your system via other, less restrictive services.

In general, only usernames belonging to real persons (i.e. excluding those such as root, uucp etc.) should be allowed to login via FTP.

Risk Rating

Medium. (If powerful usernames are omitted from this list)

Recommended Action

Ensure that powerful usernames and those not assigned to specific people are prevented from logging in via FTP, by including them in the file /etc/ftpusers. These usernames include root, bin, uucp, news, nobody, daemon and all vendor-supplied usernames.

You should also check the version of your FTP server software and consider upgrading to the latest version.

Consider the benefits of using a file quota system on user ftp, to reduce the risk of 'denial of service' attacks on your system.

26 . Other Considerations

Filtering

Ensure that only those services required from outside your domain are allowed through your routers.

Finger

Consider disabling *fingerd* or reducing the information it provides about your host.

FTP

Passwords for ***FTP*** are transmitted over the network 'in the clear', which exposes you to the risk of passwords being compromised by 'sniffer' software.

Ensure you are using the latest available software version and have applied all available ‘patches’.

Ensure you use file /etc/ftpusers. This file allows you to define those users who are NOT allowed to connect to your ftpd.

Do not keep a copy of your real /etc/passwd file as ~ftp/etc/passwd. Instead, create one from scratch, and ensure it is owned by root and has permissions 400.

Ensure file ~ftp/.rhosts does not exist.

Ensure no files or directories are owned by the ftp account or have the same group as the ftp account. These files could be replaced with a 'Trojan Horse'.

Guest Accounts

Disable guest and vendor-supplied accounts if they are not needed.

NFS

Use file /etc/exports (/etc/dfs/dfstab on some systems) to export only those file systems you want to export. Ensure file systems are exported read only whenever possible.

Remember that you are trusting security of any NFS server you use. Disable if not needed.

Apply all available patches to the NFS software.

Password Shadowing

Install and enable password shadowing if you have not already done so.

Check for unauthorised additions on a periodic basis.

Path Variable

Ensure that the path variable does not have a '.' at the beginning. This tells the system to search the current directory for issued commands before it searches the other directories in the list.

Permission Lists

Permission lists determine the type of access a user has to a file or directory.

Permissions for a file or directory are in the format TOOOGGGWWW, where:

T = file type (e.g. d = directory, ‘-‘ = plain file, l = symbolic link)
OOO = permissions for the object's owner (owner rights)
GGG = permissions for the group's members (group rights)
WWW = permissions for all other users (world rights)

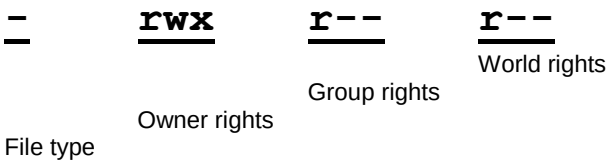
OOO, GGG and WWW are in the format rwx, where:

r = read permission (4)
w = write permission (2)
x = execute permission (1)

A '-' in the place of 'r', 'w', or 'x' indicates that the user does not have that particular permission.

SUID programs are indicated by an 's' (or 'S') in position 'x' of permissions for the file's owner. SGID files are indicated by an 's' (or 'S') in position 'x' of permissions for the file's group

A lower-case 's' indicates SUID (or SGID) *and* execute permissions. An upper-case 'S' signifies SUID (or SGID) permission, *without* execute permission.

An example of a file permission is: 

Policies & Standards

If you regularly find inconsistencies in security profiles and access rules, it is most probably due to a lack of formalised policies and standards or to a lack of clearly assigned responsibility for system ownership and security administration.

A permanent and lasting solution to security can only be achieved if these underlying issues are properly addressed.

'r' Commands

'r' commands, such as rlogin, rsh, and rexd have been a regular source of attacks in the past and should be disabled unless specifically required. Although this will increase the risk of passwords transmitted over the network being compromised by 'sniffer' software, this is probably the lesser of the two risks.

REXD

Consider disabling this service unless it is absolutely required. rexd servers generally contain little or no security and can be exploited by intruders.

Root Account

Ensure that the password for the root account is restricted to only 2 or 3 people and make it a standard to 'SU' to root, rather than logging in as root. These practices help to maintain accountability over the actions performed by the root account.

Use absolute path names to reduce the risk of executing an invalid version of a command.

Do not login over the network with root due to the risk of 'sniffer' software obtaining the password.

Ensure root does not have a ~/.rhosts file.

rlogind

Provides a remote terminal service similar to telnet. Does not require that the user type his username because it is automatically transmitted at the start of the connection.

Also, if the connection is coming from a trusted host or user, the user is logged-in without having to enter a password.

rshd

Allows a user to run a single command on a remote system. Similar concerns to rlogind, except that rsh will only work from a trusted host.

Sendmail

Use the latest version and patches. Older versions generally contain known security holes.

telnet

Allows terminal access to a remote system. When a user logs in via telnet, the user's password is transmitted unencrypted (i.e. in the clear). This presents a risk of a user's password being compromised by a 'sniffer' program on the network that listens in and collects data packets.

Trivial FTP

Disable tftp , by commenting it out from file /etc/inetd.conf. Use ftp instead.

Trusted hosts and users

A user who connects with rlogin or rsh from a trusted host can login to your system or execute a command on your system without entering a password. This can be done as long as the person uses a username that exists on both systems. The implication here is that the security on your system is very dependent on the quality of security maintained on trusted host machines.

The issues for trusted users are similar, except that the privileges mentioned above are granted to specific usernames, rather than to all users on an entire machine.

UUCP

Consider deleting the UUCP sub-system unless it is absolutely required. Otherwise, configure the uucp account with limited permissions.

Ensure the .rhosts file in the UUCP home directory is removed. Ensure no UUCP-owned files are World-writeable.

Wheel Group

Ensure you are using this facility if available on your system. It allows you to control which users can 'SU' to the root account.