This file is intended to hold general information about the audit system. It is not a tutorial but a file intended to help prevent folks from going down the wrong paths.

When I started implementing the audit support for the project I was involved in the biggest problem I had was in understanding what the audit system was auditing. The various information that was being read did not make sense. And of course I started out working with an assembler based operating system (called OS/32) and wrote a library of "C" access routines to services supplied by the operating system.

So to give folks a heads up when trying to understanding the rules and watches here is some of the problems I was having.

> The first is I had become so used to to using the utilities like chmod, chown, and the others that I forgot that the utilities were just tools to access services being supplied by the operating system.
>
> The second problem was that I was seeing the forest and not the trees. Of course since I was having to work with several different Linux distributions and Solaris that might be excusable.

And not all of the information I was finding was useful. Especially, since I was also trying to insure I was meeting some external requirements for Information Assurance to insure system integrity.

The basic rules for the audit system can be thought of as two types – a watch for files and directories and system call events for services requested from the operating system or events that occur. There is also a set of control rules.

So to monitor the user control files the following watches would be set in the rules:

```
-w   /etc/group      -p wa   -k users
-w   /etc/passwd     -p wa   -k users
-w   /etc/shadow    -p wa   -k users
```

The "-k users" is used to associate the accesses together when generating reports. What the rules listed above do is to monitor the files for any writes and attribute changes.

The rule to create access or action monitoring has the following basic format:

```
-a exit,always     -S creat -F  exit=-EACCE
-a exit,always     -S creat -F  exit=-EPERM
-a exit,always     -S creat -F  success=0
```

According to the documentation the basic layout of these access or action rules would be

```
-a <action>,<list>  -S <syscall-id>  -F <field>=<value> -k <keyname>
```

The <action> field can be "always" or "never" and the available lists are "task", "entry", "exit", "user" and "exclude". The "entry" list has been flagged as being deprecated in the future.

The auditctr man page lists various information and other rule information.

The other sources for the syscall names are “man syscalls” and even /usr/include/bits/syscall.h to get an idea of the names of the available system calls. Also, /usr/include/asm/unistd_32.h and /usr/include/asm/unistd_64.h. But remember, these are for reference information only. It is recommended that one create a test system for creating the rules and doing basic testing.

Within the audit source directory lib/ one can also examine the code that is used to generate the name used to identify the names of the system calls. In particular the following files are used to generate that tables of system call names for the various platforms:

alpha_table.h
armeb_table.h
i386_table.h
ia64_table.h
ppc_table.h
s390_table.h
s390x_table.h
x86_64_table.h

The files in Linux source directory arch/ are also useful for reviewing the available system calls that can be considered for auditing.

Again, this file is intended to help with getting started.