SE Linux MLS/MCS

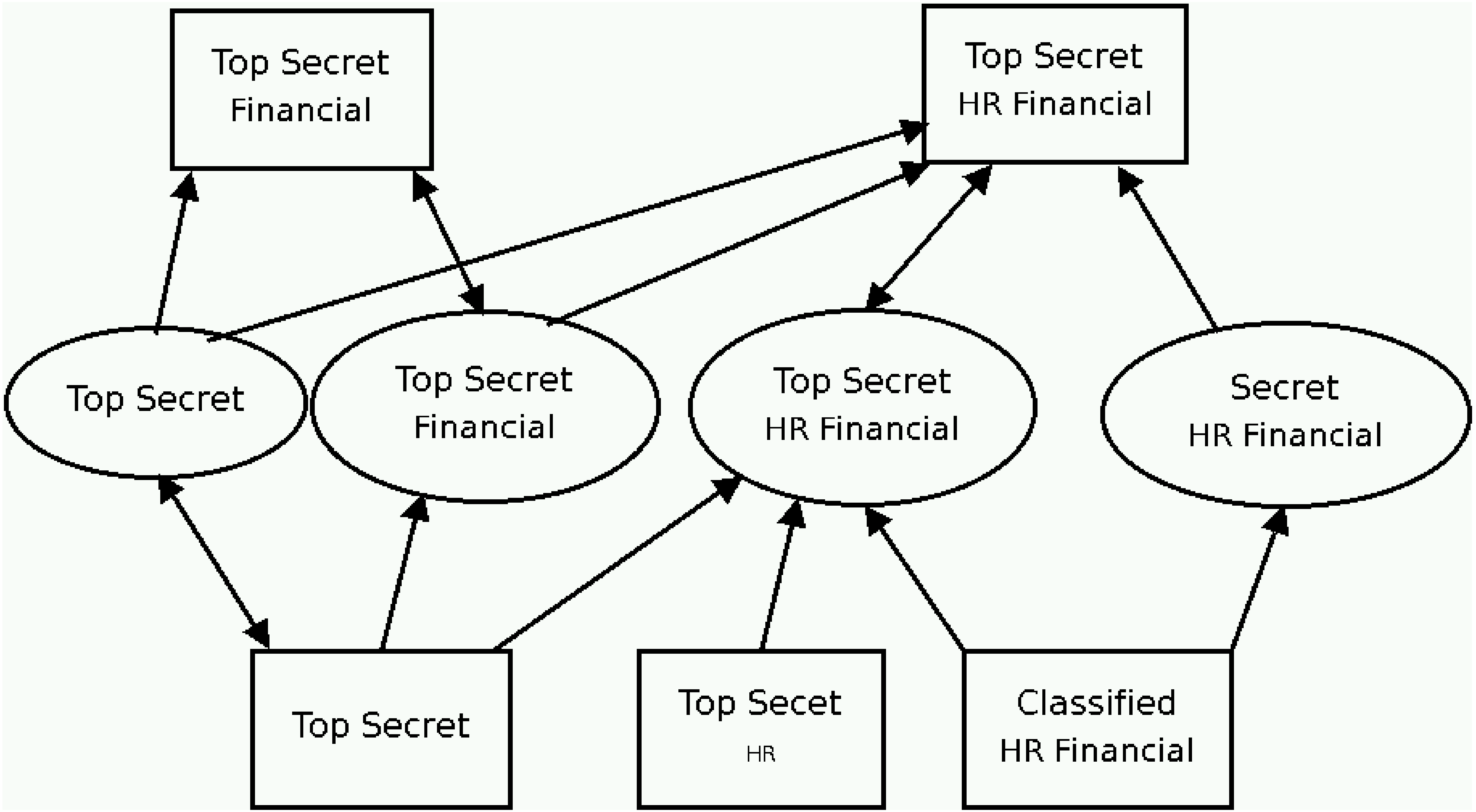
Russell Coker <russell@coker.com.au>
Internet and Security Consulting

I prefer an interactive format for my presentations, please ask questions at any time.

Multi Level Security

- Implemented in a flexible manner which is under the control of policy
- Expected that the DT model protects the system integrity while MLS protects data secrecy (no BIBA)
- MLS support includes classifications (equivalent to Top Secret, Secret, Classified, and Unclassified), default 16 numbered s0 to s15 with s15 as the most sensitive
- Also includes categories such as for departments, projects, etc. Categories are also numbered and there may be an arbitrary number of them (default 256) from c0 to c255.
- Support for preventing "read-up" and "write-down"
- A process may have a "range" from a low level to a high level, then things get complex and complicated

Multi Level Security



Multi Category Security

- The MLS code in SE Linux is very configurable. One of the many configurations of it is known as MCS.
- MCS uses categories (default 256) to label files. However in the most common configuration there is no "read up" vs "write down" distinction. If the categories possessed by a process is a super-set of the categories assigned to a file then the process will be granted full access to it.
- A process may execute a child process with a different MCS sensitivity label, but it must have less access than the parent (IE the high level must not be greater)

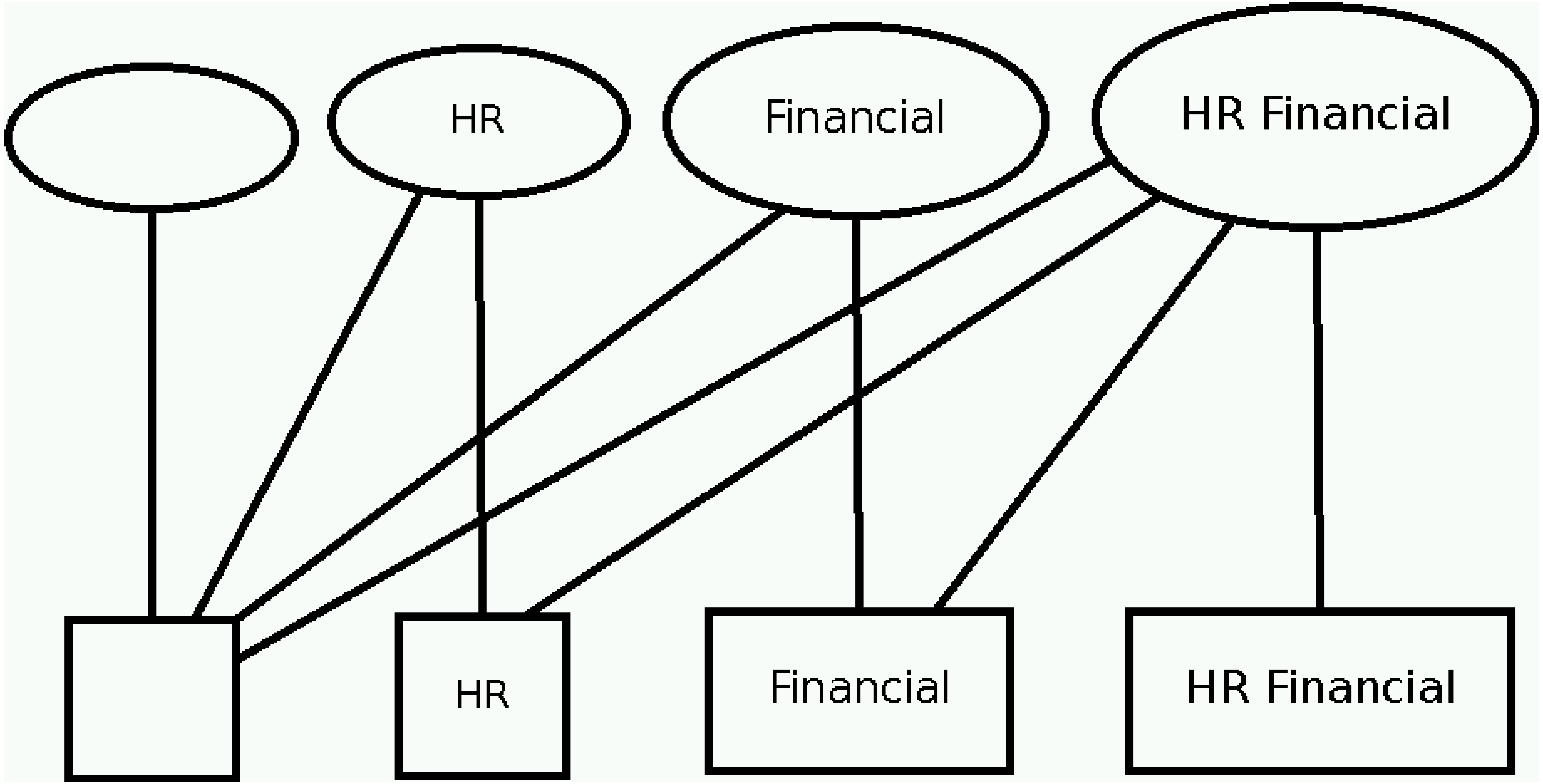
Mandatory MCS

- The process has a range denoted by it’s low level and high level. The high level determines what read access is granted as previously described. The low level determines the write access. To write to a file it’s level must be equal or greater than the low level of the process.

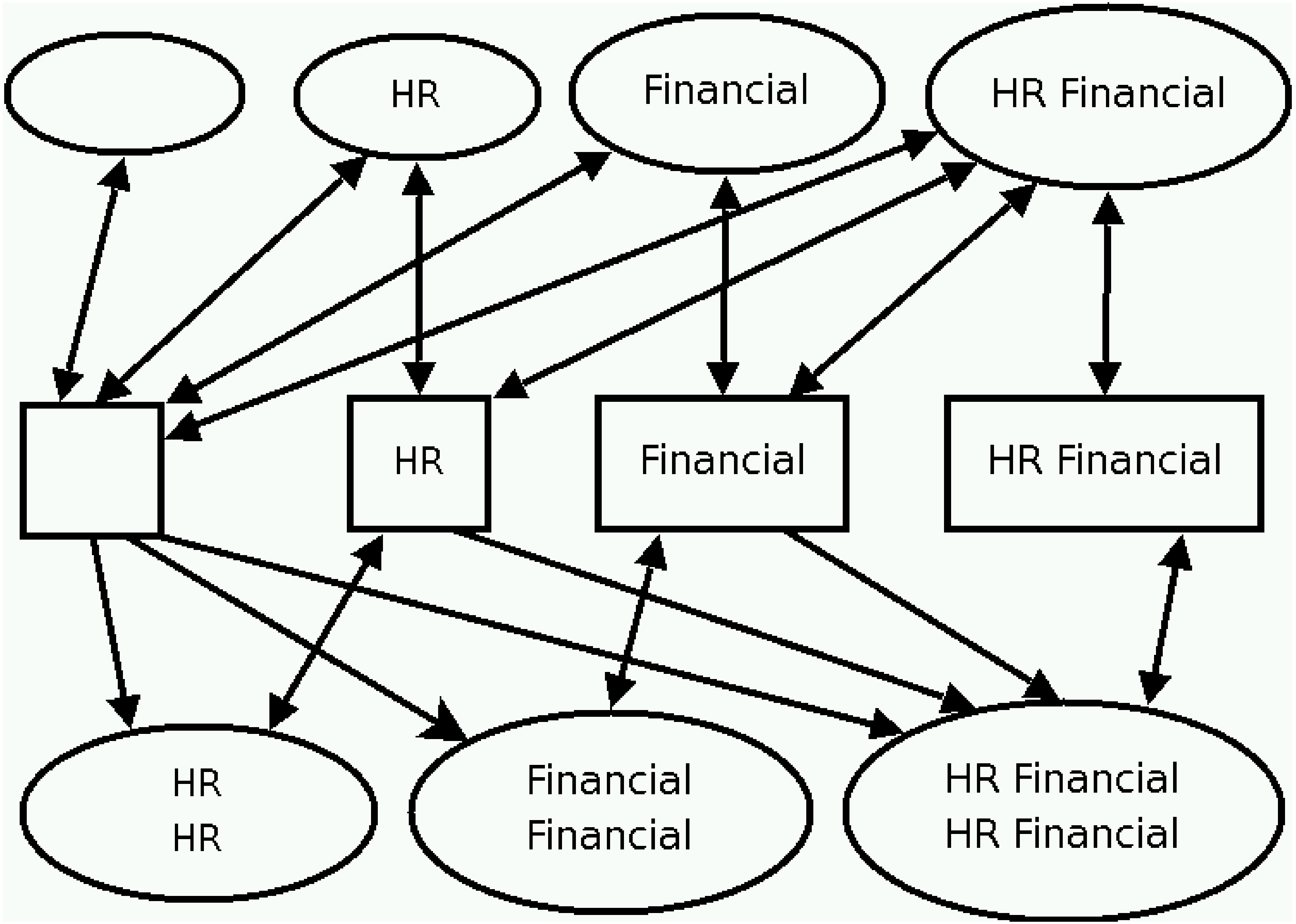
- A process may execute a child process with a different MCS sensitivity label, but it must have less access than the parent (IE the high level must not be greater and the low level must not be less)

- A process may at any time launch a child with a high level that’s less than it’s own high level or a low level that’s greater than it’s own low level as long as the high level is greater than or equal to the low level

Multi Category Security



Mandatory MCS



MCS Uses

- Could have one category per project, one category for each NDA, or other secrecy agreement.
- Users are permitted to run processes with fewer categories, so an unprivileged user who wants to run untrusted code can do so without risking it compromising their secret data.
- Can be used by the administrator to separate groups of users more effectively than Unix permissions allow (EG preventing processes being seen in ps output).

Types of Policy 1/2

- There are two variations of the Domain-Type part of the SE Linux policy in the reference policy, "strict" and "targeted".
- There are two uses of the MLS kernel features, known as "MLS" (for full Bell la Padula support) and "MCS".
- The policy variants used in Fedora Core 5 are "targeted" (targeted domain-type policy with MCS), "strict" (with MCS), and "mls" (full Bell la Padula MLS support with the strict domain-type policy).
- MLS with the targeted variant of the domain-type policy would not make sense. Great effort on protecting confidentiality with little effort on protecting system integrity would be a waste of time.
- It is possible to compile the reference policy without MLS or MCS in both the strict and targeted versions. No-one has yet considered that to be worth doing.

Types of Policy 2/2

- The file /etc/selinux/config file has options to specify whether the system should be in enforcing or permissive mode at boot time, whether SE Linux functionality should be disabled, and what type of policy should be used (strict, targeted, or MLS)

- To change between policy types on Fedora or RHEL run system-config-securitylevel in GUI mode and select the SELinux tab (NB this functionality is not available in text mode)

- $ /usr/sbin/sestatus
  - SELinux status: enabled
  - SELinuxfs mount: /selinux
  - Current mode: enforcing
  - Mode from config file: enforcing
  - Policy version: 20
  - Policy from config file: targeted

Translations

- The semanage utility is used to configure the mapping of sensitivity labels to a human readable form

- Future plans include a translation daemon which would not reveal the human-readable names for security labels to any process which does not have and equal or greater level

MCS Default Translations

- # semanage translation -a -T test s0:c1
- # semanage translation -l
  - Level Translation
  - s0
  - s0-s0:c0.c255 SystemLow-SystemHigh
  - s0:c0.c255 SystemHigh
  - s0:c1 test

MLS Default Translations 1/2

s0=SystemLow
s15:c0.c255=SystemHigh
s0-s15:c0.c255=SystemLow-SystemHigh
s1=Unclassified
s2=Secret
s2:c0=A
s2:c1=B
s0-s1=SystemLow-Unclassified
s1-s2=Unclassified-Secret
s1-s15:c0.c255=Unclassified-SystemHigh

MLS Default Translations 2/2

s0-s2=SystemLow-Secret
s0-s2:c0=SystemLow-Secret:A
s0-s2:c1=SystemLow-Secret:B
s0-s2:c0,c1=SystemLow-Secret:AB
s1-s2:c0=Unclassified-Secret:A
s1-s2:c1=Unclassified-Secret:B
s1-s2:c0,c1=Unclassified-Secret:AB
s2-s2:c0=Secret-Secret:A
s2-s2:c1=Secret-Secret:B
s2-s2:c0,c1=Secret-Secret:AB
s2-s15:c0.c255=Secret-SystemHigh
s2:c0-s2:c0,c1=Secret:A-Secret:AB
s2:c0-s15:c0.c255=Secret:A-SystemHigh
s2:c1-s2:c0,c1=Secret:B-Secret:AB
s2:c1-s15:c0.c255=Secret:B-SystemHigh
s2:c0,c1-s15:c0.c255=Secret:AB-SystemHigh

semanage - users

- # semanage user -a -L s0:c1 -r s0:c1-SystemHigh -R system_r -P user rjc
- # semanage user -l
  - Labeling MLS/ MLS/
  - SELinux User Prefix MCS Level MCS Range SELinux Roles
  - rjc user s0:c1 s0:c1-SystemHigh system_r
  - root user s0 SystemLow-SystemHigh system_r sysadm_r user_r
  - system_u user s0 SystemLow-SystemHigh system_r
  - user_u user s0 SystemLow-SystemHigh system_r sysadm_r user_r

semanage - logins

- # semanage login -a -s rjc -r s0:c1-SystemHigh rjc
- # semanage login -l
  - Login Name SELinux User MLS/MCS Range
  - __default__ user_u s0
  - rjc rjc s0:c1-SystemHigh
  - root root SystemLow-SystemHigh

semanage - ports

- # semanage port -l | head
  - SELinux Port Type Proto Port Number
  - afs_bos_port_t udp 7007
  - afs_fs_port_t tcp 2040
  - afs_fs_port_t udp 7000, 7005
  - afs_ka_port_t udp 7004
  - afs_pt_port_t udp 7002
  - afs_vl_port_t udp 7003
  - amanda_port_t tcp 10080, 10081, 10082, 10083
  - amanda_port_t udp 10080, 10081
- # semanage port -a -t http_cache_port_t -p tcp 12380
- # semanage port -l | grep 12380
  - http_cache_port_t tcp 12380, 3128, 8080, 8118

End of Part 2 - Q/A

- #selinux on irc.freenode.net
- http://www.nsa.gov/selinux/   Official SE Linux web site
- http://www.coker.com.au/selinux/   My SE Linux web pages

- Russell Coker <russell@coker.com.au>