Red Hat Linux vulnerabilities: working towards an OVAL schema

June 2002

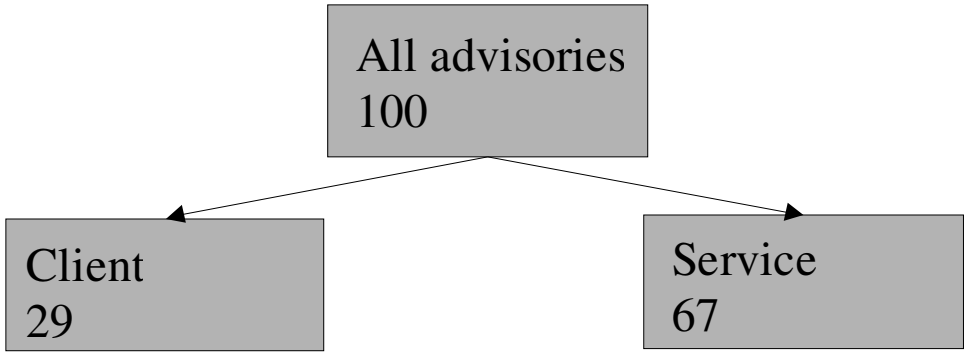
Mark J Cox, Red Hat

mjc@redhat.com

Red Hat Linux vulnerability study

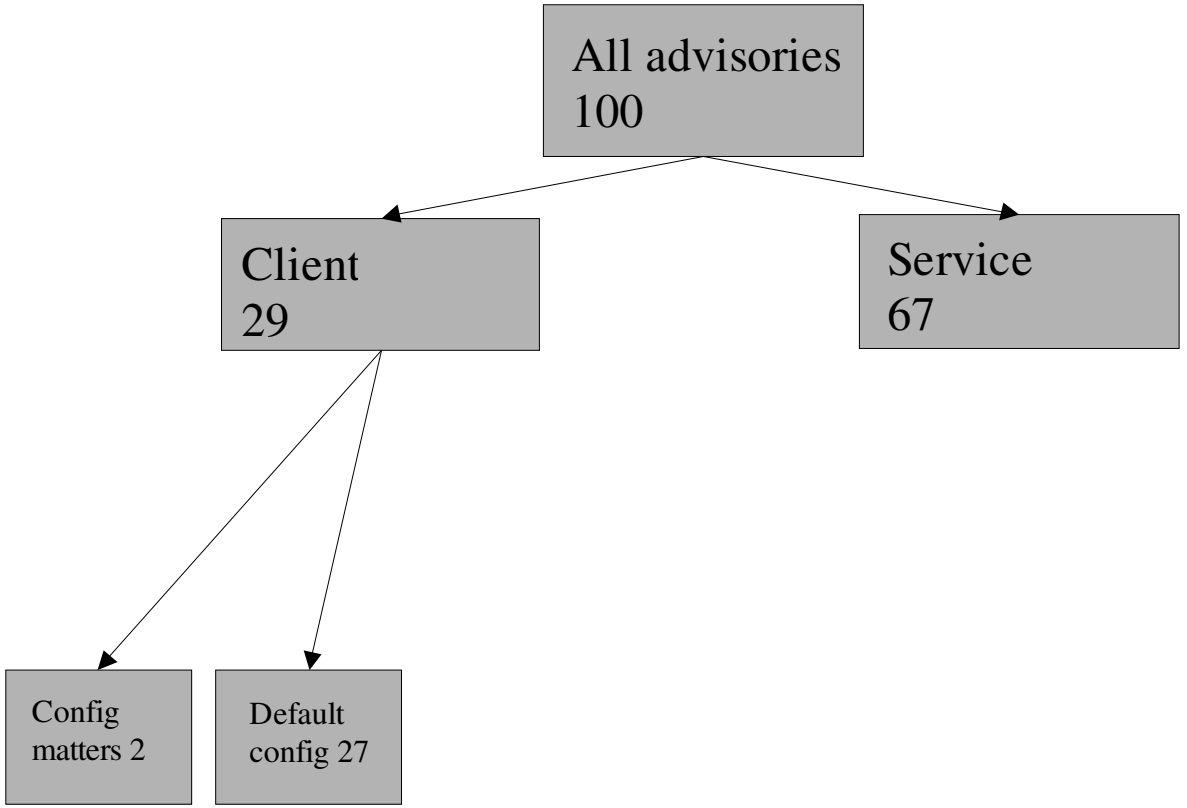
- Last 100 security advisories
  - *April 2001 to May 2002*
- Limitations
  - *Only based on RHSA not CVE name*
  - *Not taken into account what is installed by default vs what is optional*
- Definitions
  - *Client - a user has to run something to become vulnerable*
  - *Service: some inherent vulnerability or vulnerability in a 'server'*

Vulnerability breakdown



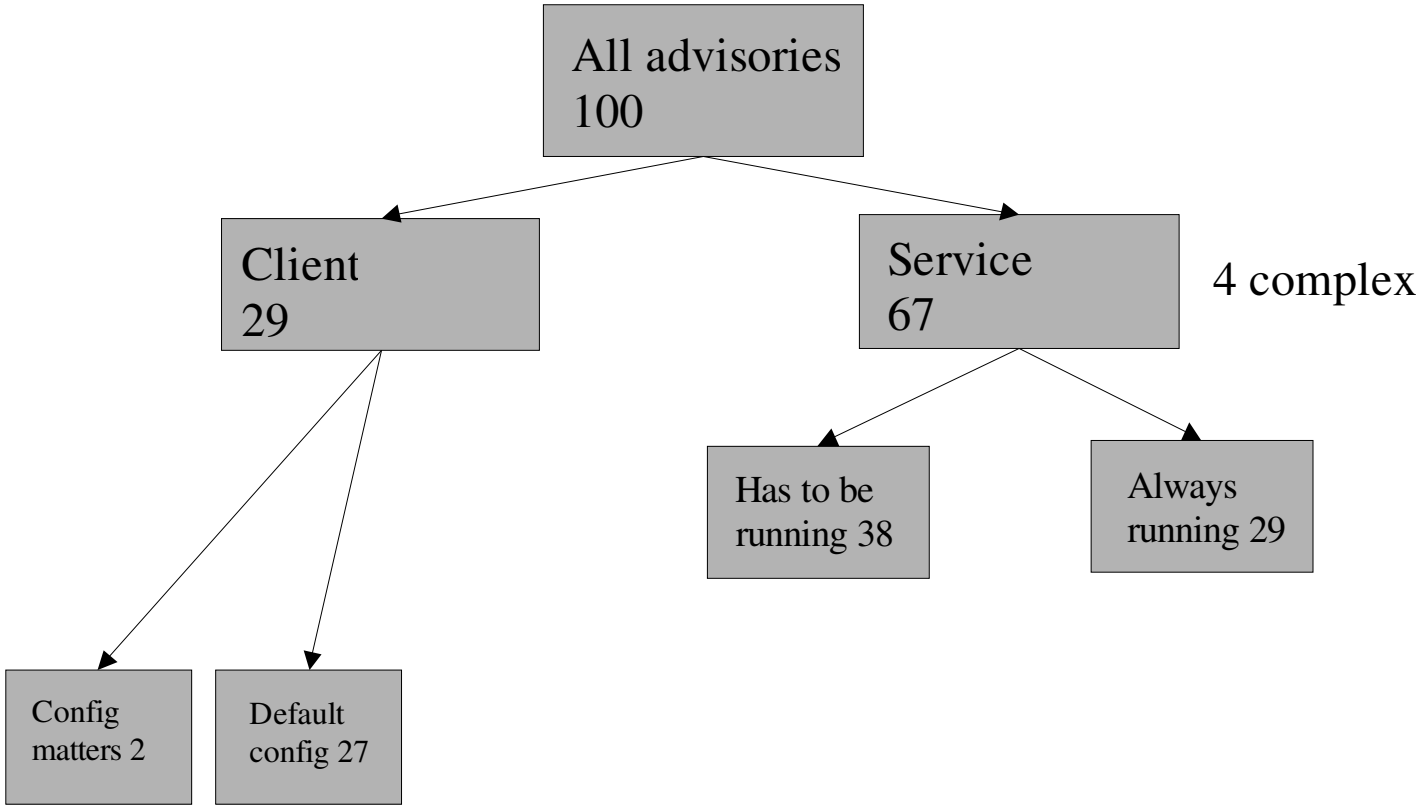
4 complex

Vulnerability breakdown

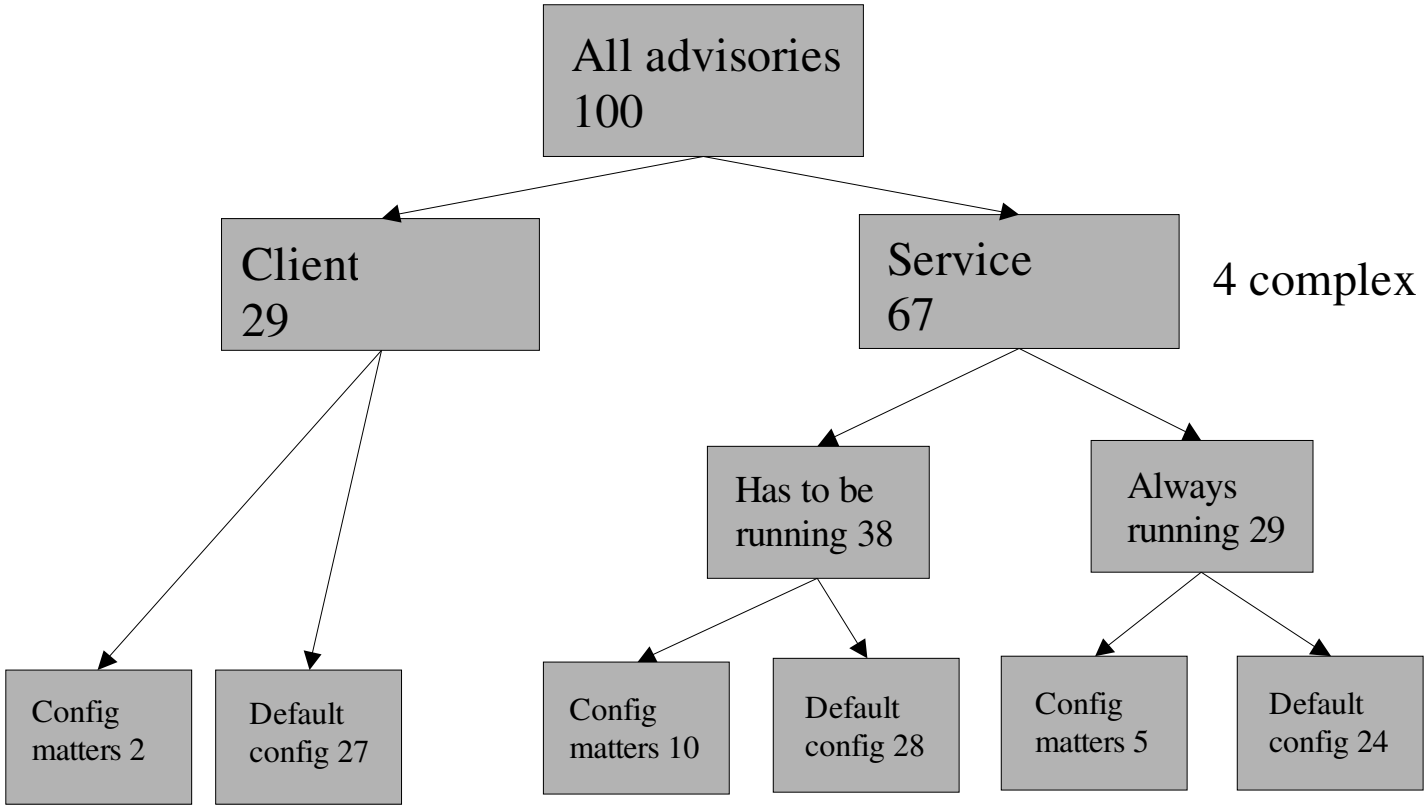


4 complex

Vulnerability breakdown



Vulnerability breakdown



Step 1: is it installed?

- Red Hat Linux uses RPM
- RPM maintains database of installed packages
  - *Easy to find out what file a package belongs to*
  - *Packages are versioned "NVR"*
  - *Contains MD5 of installed files*
  - *Can check what files have changed*
- Assume that we're checking for vulnerable software users have installed by RPM?
  - *Checking for installation is as simple as comparing as a NVR check (using >, <)*
  - *NVR versioning depends on RHL release not on actual package version*
- Sufficient for 51 of our examples

Step 2: is it running?

- Client depends on user running, no way to tell (29)
- Service may be running by default (29)
- Service running status can be found using chkconfig database
  - *knows what services are started at what runlevels*
  - *Xinetd*
- Assumes user doesn't start service by hand the wrong way
  - *("service httpd start" is the right way)*
- Sufficient for 79 of our examples

Step 3: configuration

- Remaining 17 examples are only vulnerable if a particular configuration is set
- Parsing config files is hard
  - *Parsing Apache config files is almost impossible because of .htaccess, overrides, non-XML structure*

Step 4: has it been modified?

- A user may have
  - *Recompiled with the same NVR*
  - *Manually started a service by the wrong method*
  - *Changed some file permissions*
  - *Changed some file*
- RPM can tell us this

Working towards a Schema

- Checking if a package exists with a NVR check.
- Checking if package is running with chkconfig
- Checking if a package has been modified
  - *Perhaps just identify the critical bits?*
- ... Isthe best we can do for 79 of the examples
- Notes
  - *Each CVE name may require doing these checks on more than one package.*
  - *Use of RPM means we don't need file level checks like other schema*

Potential outputs

- You are vulnerable to X
- You may be vulnerable to X
  - *It depends on your configuration (default config?)*
- You are not vulnerable to X
  - *Because you don't have the package installed*
  - *Because the package isn't running*
- We don't have a clue if you are vulnerable
  - *Because you have messed with something*