CFEngine on “odd” Platforms

CFEngine

- What I am considering “odd” platforms
- A bit of history
- What the “core” knows
  - Processes
  - Users
  - Mounts
- What Masterfiles Policy Framework knows
  - Services
  - Packages
- Platform specific quirks
  - Yocto
  - Termux (Android)
  - Alpine (PostmarketOS)
  - OpenBSD
  - iSH (iOS)

Supported Platforms

We currently support in both Enterprise and Community packages:

- Debian 9, 10 and 11
- RedHat/CentOS 7, 8 and 9
- Ubuntu 16, 18, 20 and 22
- SuSE 11, 12, 15
- Windows Server 2008, 2012
- Solaris 11
- HP-UX 11.31+
- AIX 7.1 TL5, 7.2

Plus a new architecture: aarch64 for Debian 11 and Ubuntu 22.

CFEngine

Odd Platforms

Those which I will discuss today:

- Yocto
  - Needs an upgrade from 3.15.0 in meta-openembedded
- Termux (Android)
  - Up-to-date package, needs policy fixes
- Alpine (PostmarketOS)
  - Up-to-date package, needs policy fixes
- OpenBSD
  - Very outdated 3.7.2 in ports
- iSH (iOS)
  - Truly an older alpine, 3.14, binaries broken

A bit of history…

As of January 9, 2009 near the beginning of CFEngine 3 the core code had knowledge of the following platforms:

sun4, ultrix, hpux, aix, linux, solaris, osf, digital, sun3, irix4, irix, irix64, freebsd, solarisx86, bsd4_3, newsos, netbsd, aos, bsdos, nextstep, cray, gnu, nt, unix_sv, openbsd, sco, darwin, ux4800, qnx, dragonfly

Currently, February 5, 2023, the core code knows about these platforms:

hpux, aix, linux, linux, solaris, sun solaris, freebsd, netbsd, cray, windows, unix_sv, openbsd, sco, darwin, qnx, dragonfly, windows, vmware, android

Currently, February 5, 2023, we natively know about these platforms:

hpux, aix, linux, linux, solaris, sun solaris, freebsd, netbsd, cray, windows, unix_sv, openbsd, sco, darwin, qnx, dragonfly, windows, vmware, android

Processes

- Promise type
- Alpine/yocto need procps, busybox, core doesn’t detect (only for android) TODO

Users

- Promise type
- Systemd services
- Standard services (sysvinit)
- implementation uses either useradd or pw (freebsd)
- Change password

Services

- Promise type
- Systemd services
- Standard services (sysvinit)

CFEngine

Services

- Promise type
- Systemd services
- Standard services (sysvinit)

CFEngine

Packages

- Promise type
- Package modules
- Package methods (deprecated but interesting!)

Package Modules Then

April 10, 2014, version 3.7 packages.cf:

- debian (apt)
- redhat (yum)
- suse (zypper)
- freebsd (pkg)
- alpine (apk)
- gentoo (emerge)
- archlinux (pacman)

Package Modules Now

packages.cf currently has knowledge of:

- debian/termux(android) (apt)
- redhat/amazon linux (yum)
- suse/sles/opensuse (zypper)
- Aix (nimclient)
- Slackware (slackpkg)
- Windows (msiexec)
- Alpinelinux (apk)
- freebsd (pkg)
- alpine (apk)
- gentoo (emerge)
- archlinux (pacman)

Package Methods

- Pip
- Npm (local or global)
- Brew (as user)
- Windows Feature
- Yum group
- Ips (open solaris)
- Smartos (solaris 10 fork)
- Opencsw (solaris)
- Freebsd portmaster

- What I am considering “odd” platforms
- A bit of history
- What the “core” knows
  - Processes
  - Users
  - Mounts
- What Masterfiles Policy Framework knows
  - Services
  - Packages
- Platform specific quirks/issues
  - Yocto - /var/libntech
  - Termux (Android) - odd PREFIX, no root (by default)
  - Alpine (PostmarketOS) - openrc services, partial sysvinit
  - OpenBSD - no users promises
  - iSH (iOS) - execution problem probing for network interfaces