WebKit

JIT?

I knew JIT was disabled, but had a hell of a time determining where in the source code that was being enforced until I back tracked ALL the way up to the top to ProcessLauncher::launchProcess() in ProcessLauncherCocoa.mm

There we can see:

- `launchProcess()`
    - `serviceName(m_launchOptions, m_client);`
        - `webContentServiceName(launchOptions.nonValidInjectedCodeAllowed, client);`

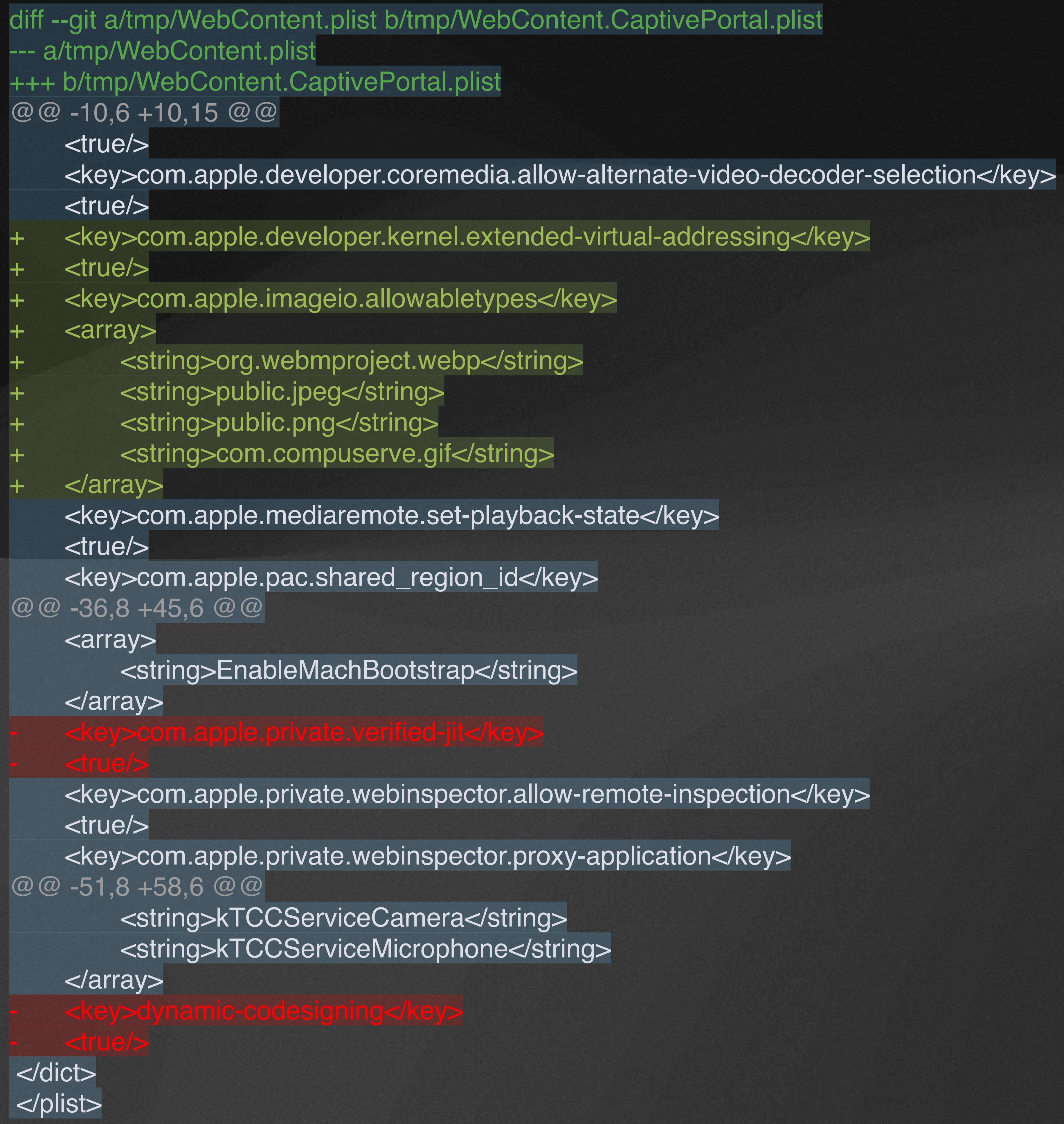
```
static const char* webContentServiceName(bool nonValidInjectedCodeAllowed, ProcessLauncher::Client* client)
{
    if (client && client->shouldEnableLockdownMode())
        return "com.apple.WebKit.WebContent.CaptivePortal";
    return nonValidInjectedCodeAllowed ? "com.apple.WebKit.WebContent.Development" : "com.apple.WebKit.WebContent";
}
```

If Lockdown Mode is enabled then the WebKit launcher will start a com.apple.WebKit.WebContent.CaptivePortal process instead of the normal com.apple.WebKit.WebContent process.

You can also see what entitlements these different services get assigned in the file Source/WebKit/Scripts/process-entitlements.sh

WebKit

What's the difference you ask?



```
diff --git a/tmp/WebContent.plist b/tmp/WebContent.CaptivePortal.plist
--- a/tmp/WebContent.plist
+++ b/tmp/WebContent.CaptivePortal.plist
@@ -10,6 +10,15 @@
        <true/>
        <key>com.apple.developer.coremedia.allow-alternate-video-decoder-selection</key>
        <true/>
+       <key>com.apple.developer.kernel.extended-virtual-addressing</key>
+       <true/>
+       <key>com.apple.imageio.allowabletypes</key>
+       <array>
+               <string>org.webmproject.webp</string>
+               <string>public.jpeg</string>
+               <string>public.png</string>
+               <string>com.compuserve.gif</string>
+       </array>
        <key>com.apple.mediaremote.set-playback-state</key>
        <true/>
        <key>com.apple.pac.shared_region_id</key>
@@ -36,8 +45,6 @@
        <array>
                <string>EnableMachBootstrap</string>
        </array>
-       <key>com.apple.private.verified-jit</key>
-       <true/>
        <key>com.apple.private.webinspector.allow-remote-inspection</key>
        <true/>
        <key>com.apple.private.webinspector.proxy-application</key>
@@ -51,8 +58,6 @@
                <string>kTCCServiceCamera</string>
                <string>kTCCServiceMicrophone</string>
        </array>
-       <key>dynamic-codesigning</key>
-       <true/>
 </dict>
 </plist>
```

So the process itself does NOT even have the entitlements to do JIT if it wanted to.

You can also see they are using entitlements to limit the kinds of images that are supported via ImageIO: webp, jpeg, png, and gif.

Sandbox

As Apple moves to SANDBOX_VERSION_3, they appear to be moving this functionality into the sandbox profile enabling it by using the LockdownModeEnabled sandbox state-flag.

I'm very curious to see what the new EnableExperimentalSandbox and EnableExperimentalSandboxWithProbability state-flag will do 👀

UPDATE: they seemed to have removed them 😞