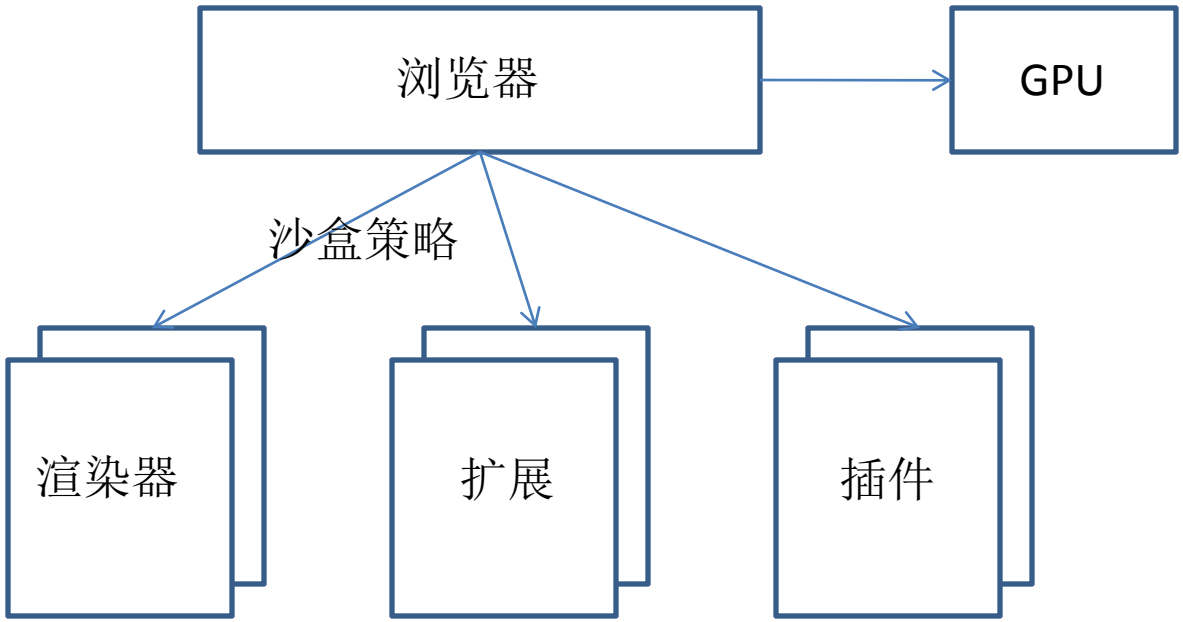
Chrome安全架构的演化

Huan Ren

Director, Qihoo 360 Technology Ltd

Chrome当前架构



历史

- 最初版本: 多进程, 无沙盒
- 2007: 渲染沙盒
- 2009: 扩展系统
- 2010: GPU渲染机制
- 2010 至今: 插件沙盒 以及 Pepper

Windows 系统上的渲染沙盒

- 令牌

  调用函数 *CreateRestrictedToken*， 参数SID为null, 删除所有权限.
- 任务

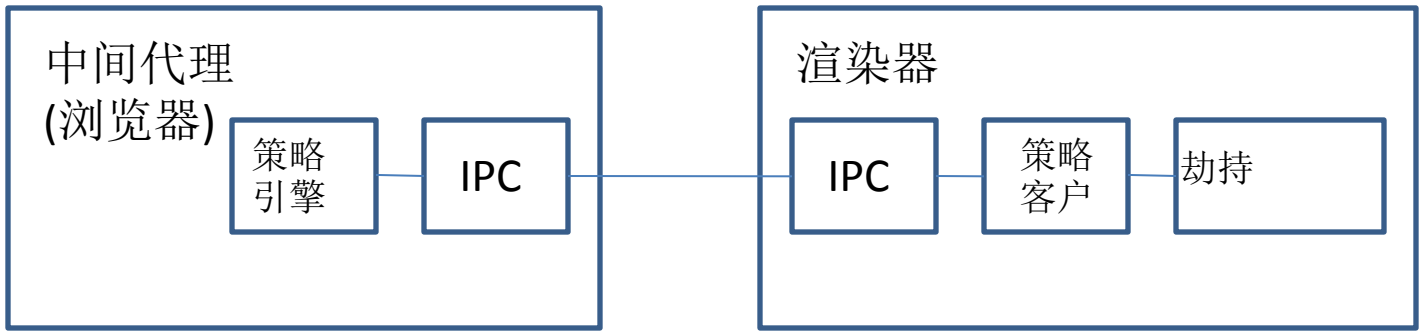
  JOB_OBJECT_LIMIT_ACTIVE_PROCESS

  JOB_OBJECT_UILIMIT_READCLIPBOARD

  …
- 切换桌面
- 低可信级(Vista+)

挑战:兼容性

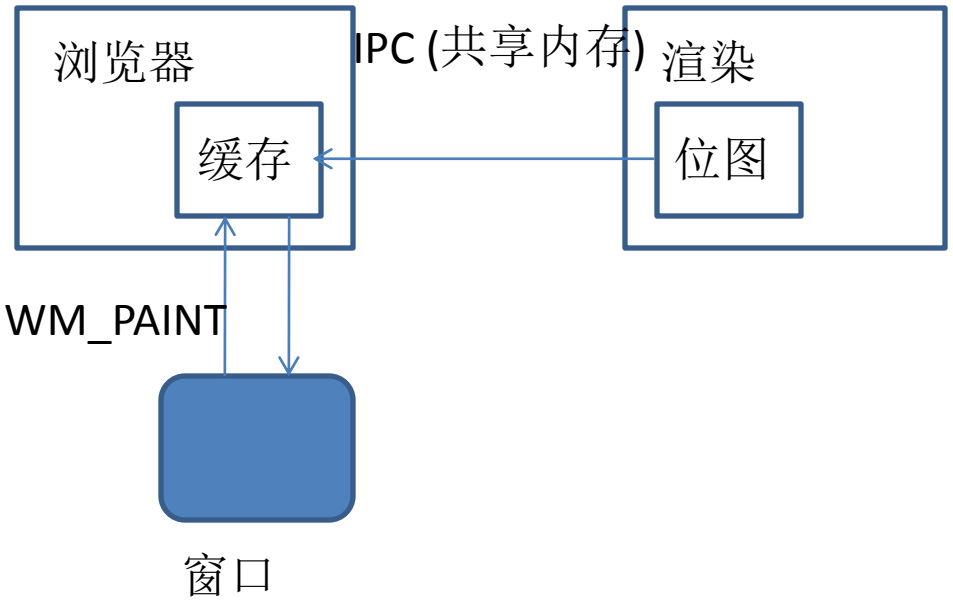
- 两个阶段
  - 启动:初始令牌
  - 锁定: 调用 *LowerToken()* 之后
- API劫持



为了兼容性而劫持API，而不是为了沙盒机制

挑战:兼容性

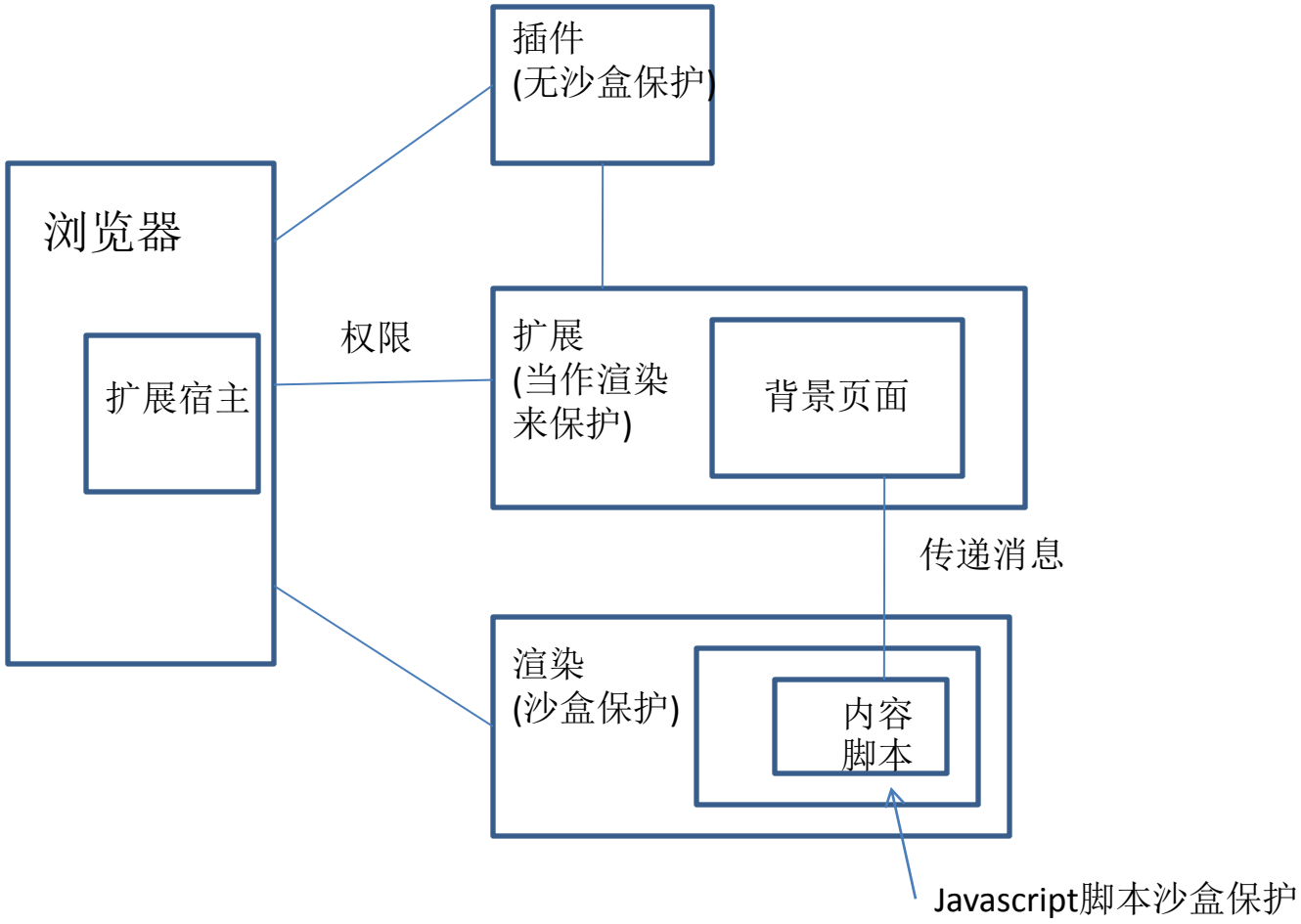
- 界面绘制



渲染进程隔离

- 进程模型
  - 每个标签一个进程
  - 每个站点一个进程：同域名范畴的网站放在一个进程
  - 每个站点实例一个进程：某个打开的网站链开的一系列网站都属于一个进程
- 强制进程隔离
  - webUI(Web页面图形界面), 扩展, 以及普通渲染进程

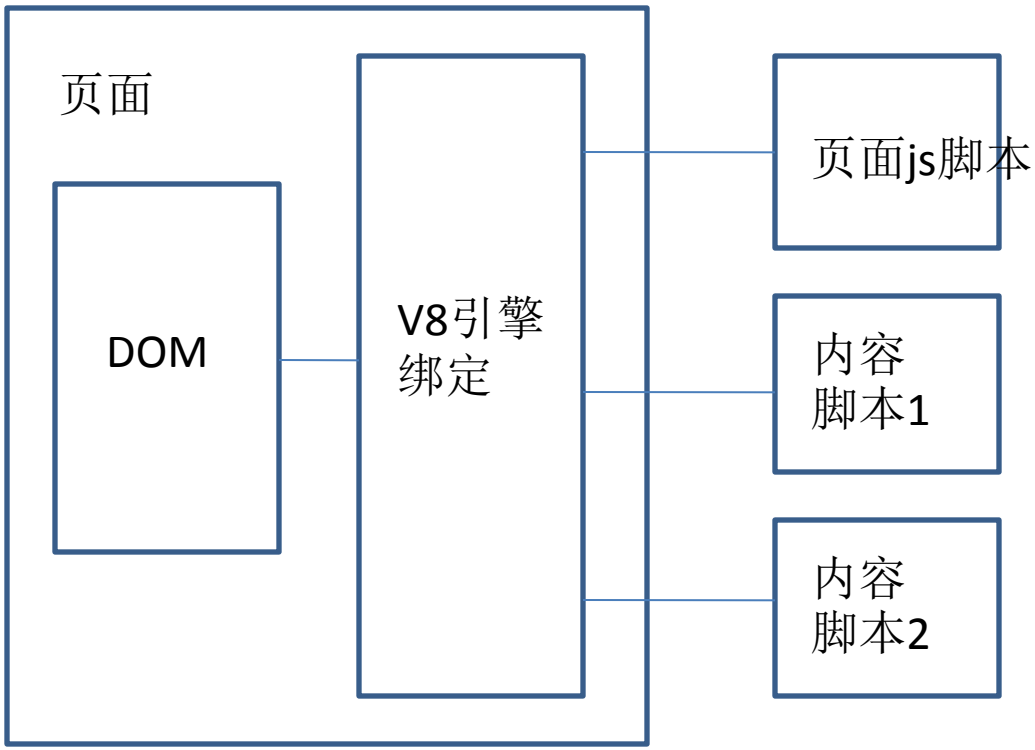
扩展机制的安全架构


插件
(无沙盒保护)
浏览器
扩展宿主
权限
扩展
(当作渲染
来保护)
背景页面
传递消息
渲染
(沙盒保护)
内容
脚本
Javascript脚本沙盒保护

扩展安全机制的元素

- Javascript脚本沙盒
- 内容脚本以及核心扩展的隔离
- 权限
- 扩展发布

Javascript脚本沙盒 :隔离的世界



权限隔离

- 内容脚本: 在页面关联的渲染进程中运行
- 扩展核心: 在单独进程中运行，拥有如下权限
  - 发出 cross-origin XMLHTTPRequest
  - 调用插件的 APIs
  - 加载插件
- 都作为渲染进程来做沙盒保护

消息传递

- 一次请求
  - chrome.extension.sendMessage
  - chrome.tabs.sendMessage
  - chrome.extension.onMessage.addListener
- 长连接
  - chrome.extension.connect
  - chrome.extension.onConnect.addListener
- 跨越扩展的消息传递

发布以及权限声明

- Manifest

```
{
    …
    "key": "publicKey",
    "permissions": [
      "tabs",
      "bookmarks",
      "http://*.google.com/",
      "unlimitedStorage" ],
    "plugins": [...],
}
```

常见插件弱点

- 网络攻击

  在 <script src> 中添加一个HTTP URL地址

- XSS

  eval(), innerHTML, document.write()

```
function displayAddress(address) {
    eval("alert('" + address + "')");
}
```

Chrome插件扩展

- UC Berkeley的研究成果, 发表在 in USENIX 安全会议 2012
  - 手动检查50个流行的以及50个任意选择的插件.
  - 40个插件中发现了70个漏洞

Source: "An Evaluation of the Google Chrome Extension Security Architecture"

Chrome插件扩展

脆弱模块	Web攻击	网络攻击
核心扩展	5	50
内容脚本	3	1
网站	6	14

脆弱模块	流行	任意选择	全部
核心扩展	12	15	27
内容脚本	1	2	3
网站	11	6	17
任意	22	18	40

Source: "An Evaluation of the Google Chrome Extension Security Architecture"

插件安全v2

- 支持内容安全策略 (Content-Security-Policy ,CSP)
- Manifest V2

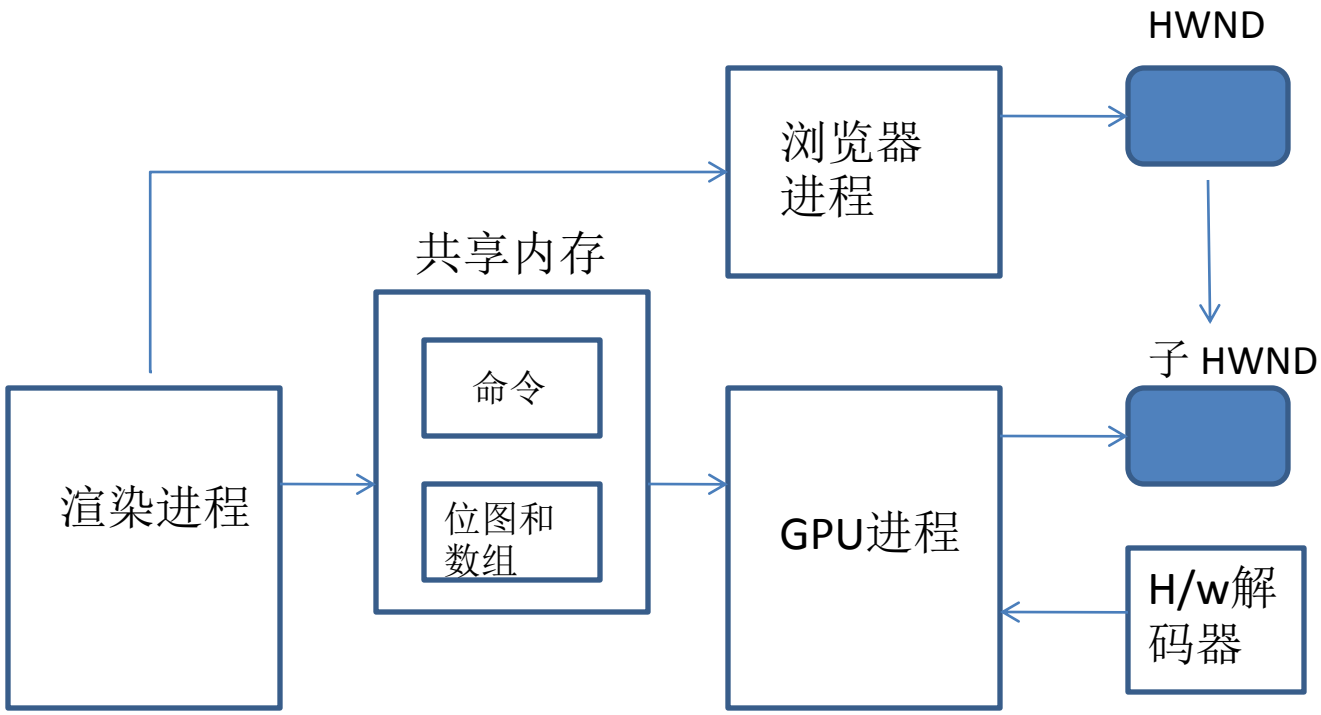
  script-src 'self'; object-src 'self'
  - 无内嵌脚本
  - 无 eval()
  - 仅仅从包内部或者白名单加载对象
- “减少 96% (49 out of 51)核心扩展漏洞. ”

扩展的其他威胁

- 威胁模型
  - 针对核心扩展的攻击
    - 主要涉及目标
  - 恶意扩展
    - Chrome更新机制进一步放大了该威胁
  - 扩展可修改页面的网站
    - 有待研究

- 恶意扩展
  - 从Chrome 21起, 只允许从在线市场安装.

GPU 进程



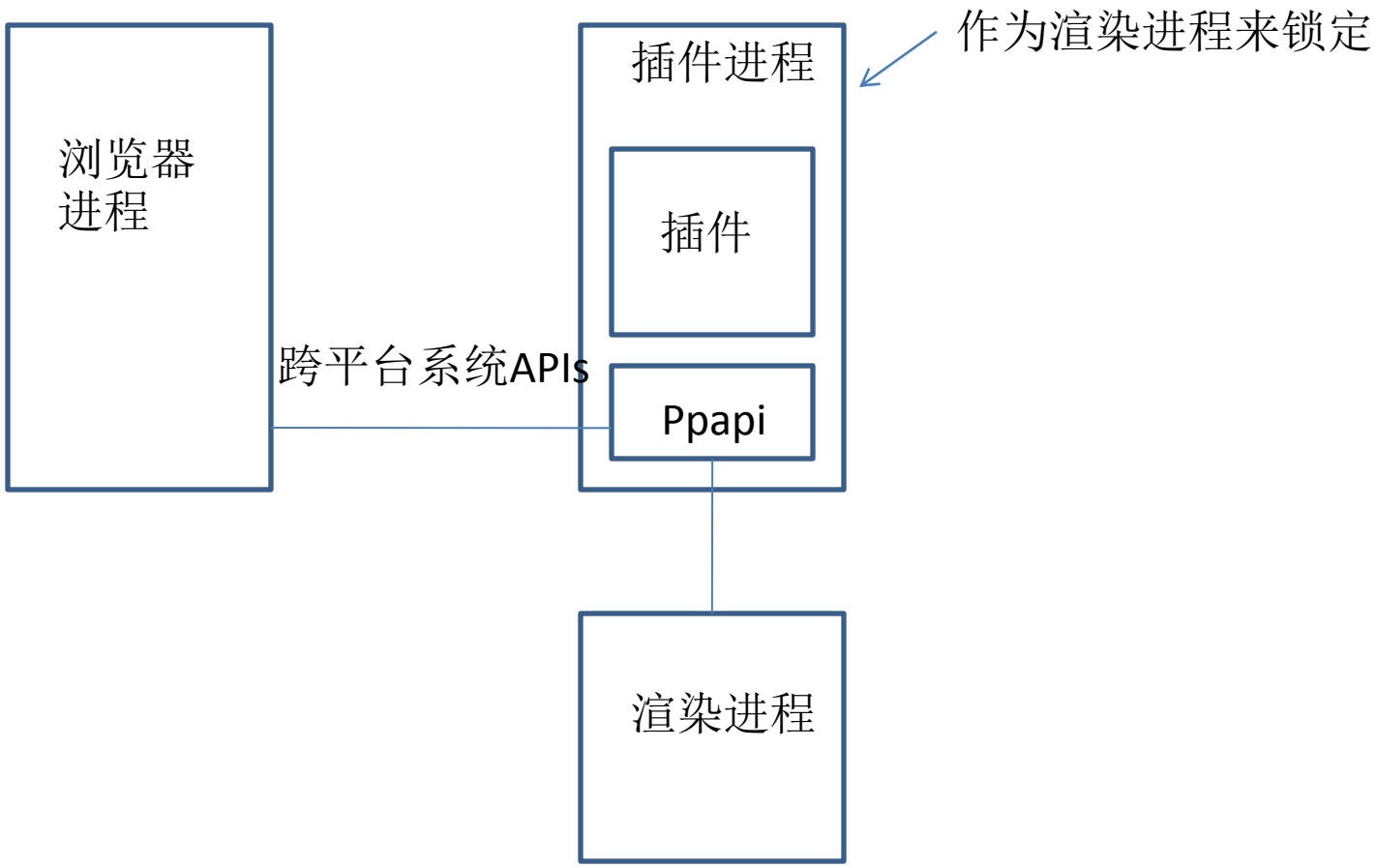
GPU Sandbox

- 令牌
  - WinBuiltinUsersSid,
    WinWorldSid,
    WinRestrictedCodeSid
- 与交互桌面连接

Plug-ins

- NPAPI插件没有被沙盒保护
  - 系统的最弱一环
- 保护措施
  - 黑名单
  - 点击播放
  - 内置的Flash播放器
    - 快速更新
    - 沙盒: Vista以后版本的系统, 低可信模式

Ppapi插件



Ppapi 示例

```
struct PPB_FileIO_1_0 {
  …
  int32_t (*Open)(PP_Resource file_io,
                  PP_Resource file_ref,
                  int32_t open_flags,
                  struct PP_CompletionCallback cb);
  …
}
```

当前进展

- 无窗口模式下取得性能改进
  - From sync layout model to async
- 转换本地系统调用为 ppapi
  - Flash
  - PDF 阅读器
- 从Chrome 21起, Ppapi Flash默认启用

设计原则总结

- 最小权限
- 权限隔离
- 利用系统内置安全机制
- 取得系统，性能以及用户体验的平衡

Chromium项目的贡献开发者

- Google
- Qihoo 360
- 个人开发者

- 给Chromium做50个补丁会让你在哪里都能得到一份工作

招聘

发送邮件到

- 360 browser

  renhuan@360.cn

- 360 Safe Guard

  paulfan@360.cn

感谢!