Survey on Behaviors of Captive Portals

Mariko Kobayashi
ao@sfc.wide.ad.jp

IETF98, Chicago
March 2017

Abstract

□ In capport BoF in IETF 97

- “We needs a volunteer for survey about capport.”

□ I implemented a survey tool kit to automate the survey(on Raspberry Pi) and use it for this time.

□ This survey shows an actual situation of Captive Portals in Japan.

Survey Overview

□ Survey on behaviors of **40** Captive Portals in **Central area of Tokyo, Japan**.

□ Survey Items:

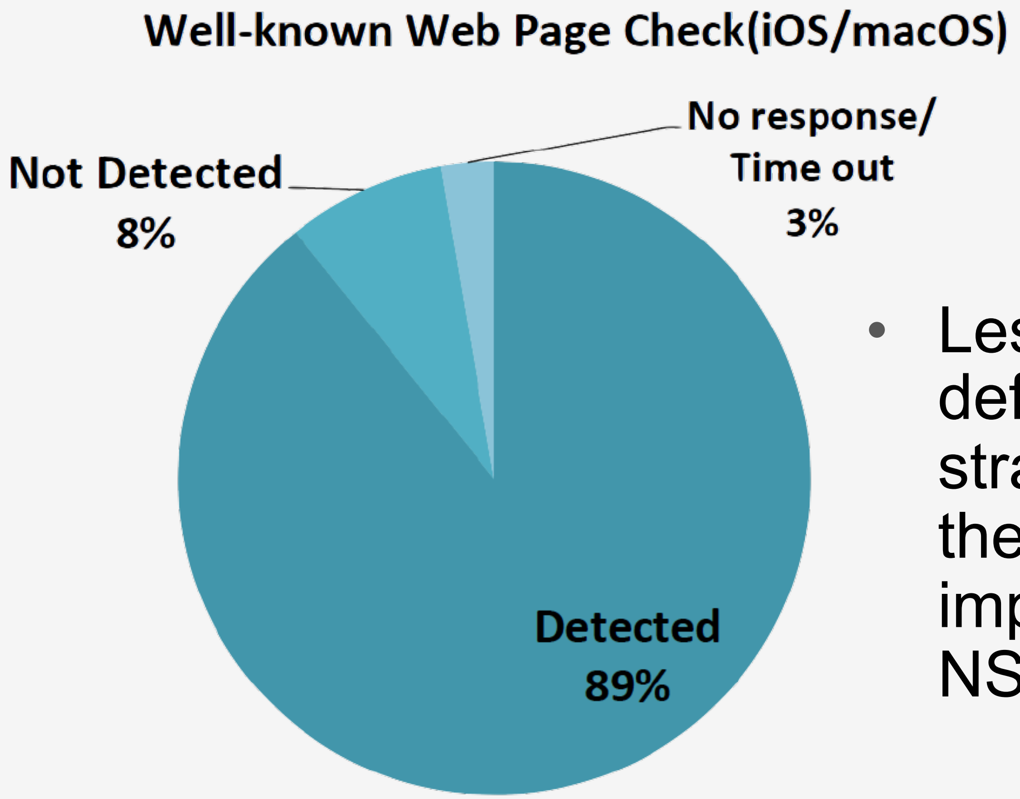
1. **False Negatives**(iOS/macOS, Windows, Android)
2. **HTTP Status Code**
3. **DNS poisoning**

Basic capport Detection Strategy

	Detection Strategy (Well-Known Web Pages)	Full Internet Access (not capport)	Captive Portal
iOS/macOS (Apple)	• Access captive.apple.com to check the Internet • connectivity	• txt ”Success”	• Cannot get txt ”Success”
Windows (Microsoft)	✓ Request DNS lookup for dns.msftncsi.com (to judge whether Captive Portal or bad internet connectivity) ✓ request http://www.msftncsi.com/ncsi.txt	• IP address 「133.107.255.255」 • txt ”Microsoft NCSI”	• IP Address 「133.107.255.255」 • Cannot get txt ”Microsoft NCSI”
Android (Google)	• Access http://google.com/gen_204 (for HTTP probe) • https://google.com/generate_204 (for HTTPS probe)	• 204 & No Content	• Cannot get both “204” and “No Content”

References =>P.16

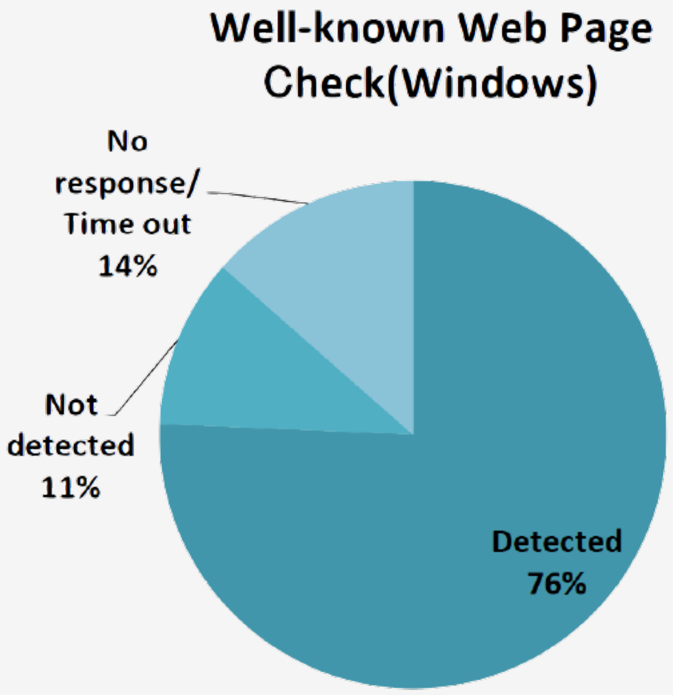
1.1 False Negatives(macOS/iOS)

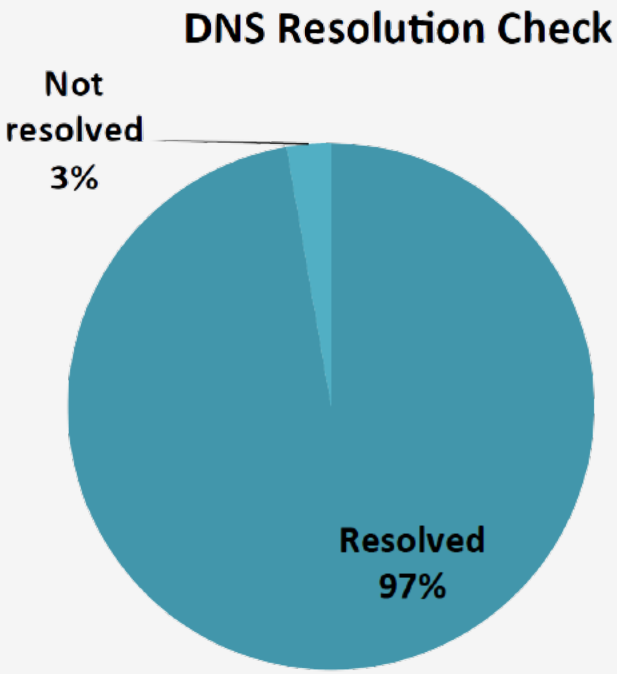


- Less than 10% of capport defeat the detection strategy and some of them had been implemented by the same NSP.

- They defeat detection for some reason…?

1.2 False Negatives(Windows)

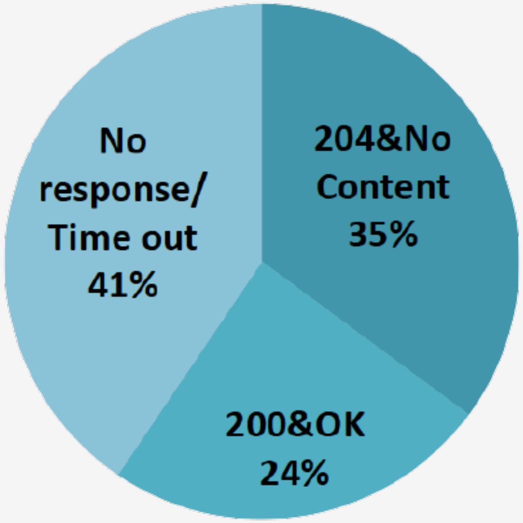




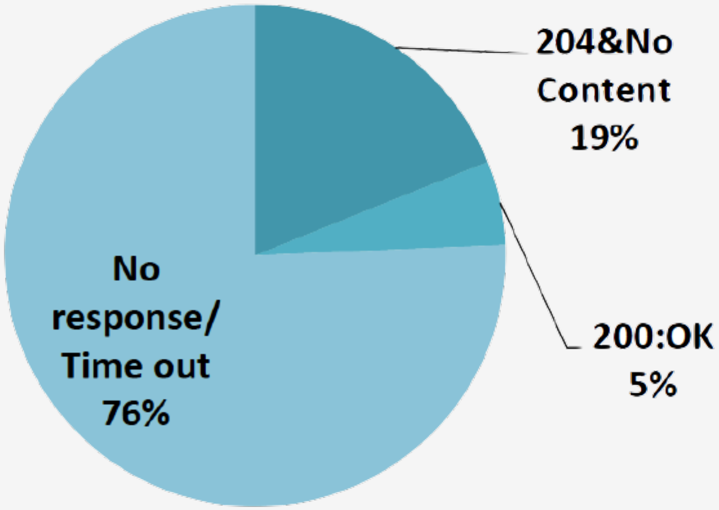
- 11% of capport defeat Window's detection strategy, but I cannot find remarkable characteristics.

1.3 False Negatives(Android)

http://www.google.com/gen_204



https://www.google.com/generate_204



□ One-third of capport defeat the detection strategy, and they replies 204 & No Content to Android’s well-known Web page(it means “not capport”)

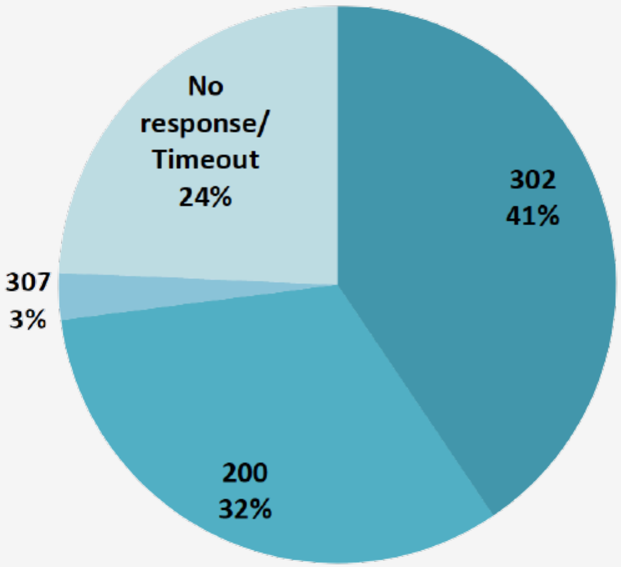
□ Most of the HTTPS probe does not success on capport.

2 Status Code

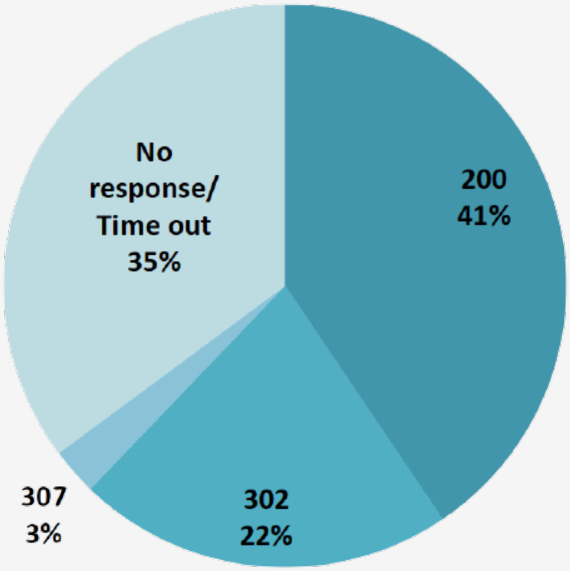
- Current Proposal : **511**?(by mnot)
- Actual status code when the terminal accesses Well-Known Web Pages(of iOS/macOS or Windows)

  : **302, 200, 307**

iOS/macOS

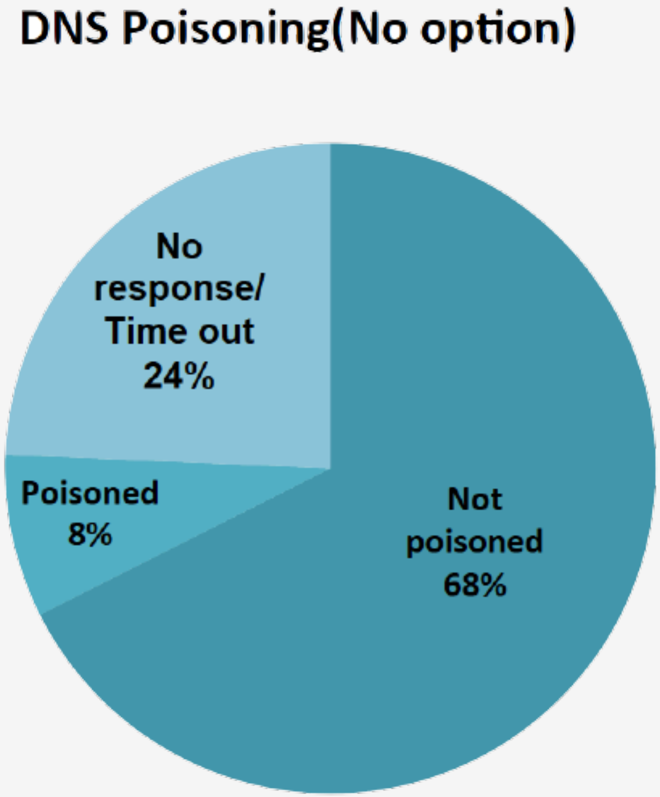


Windows



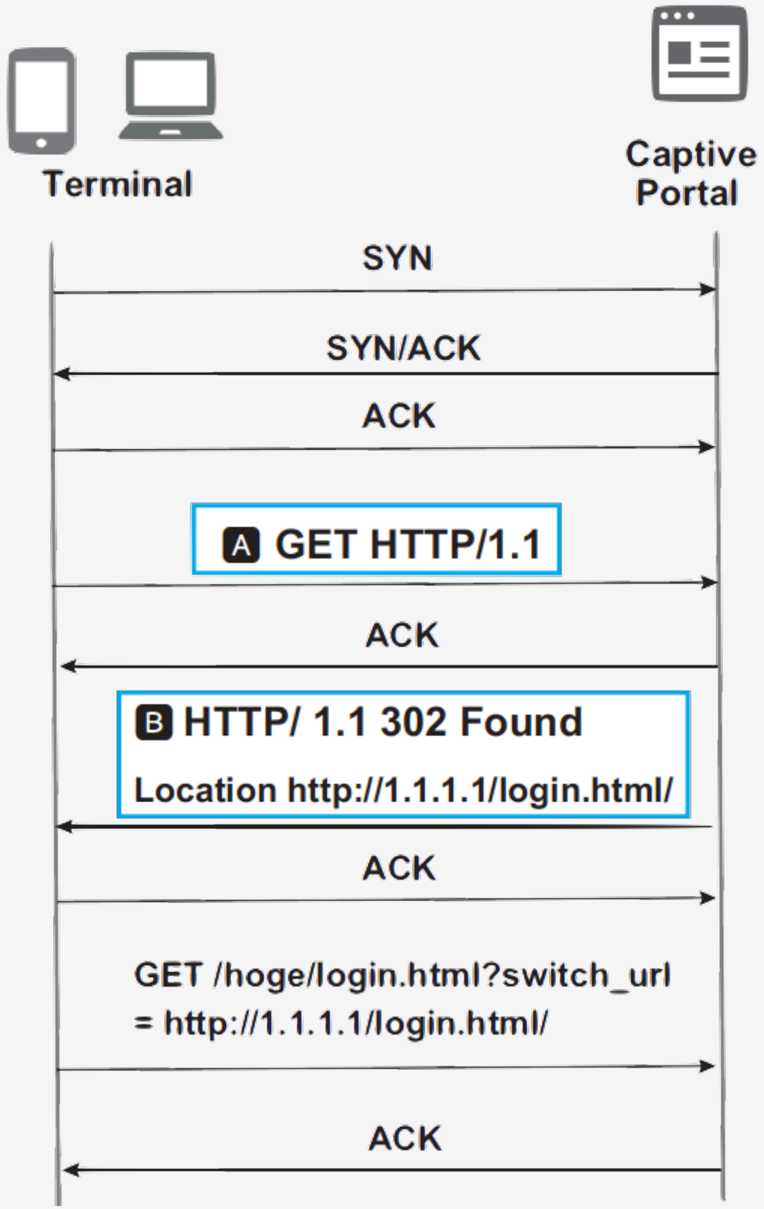
3 DNS Poisoning

□Can be detected DNS Poisoning?



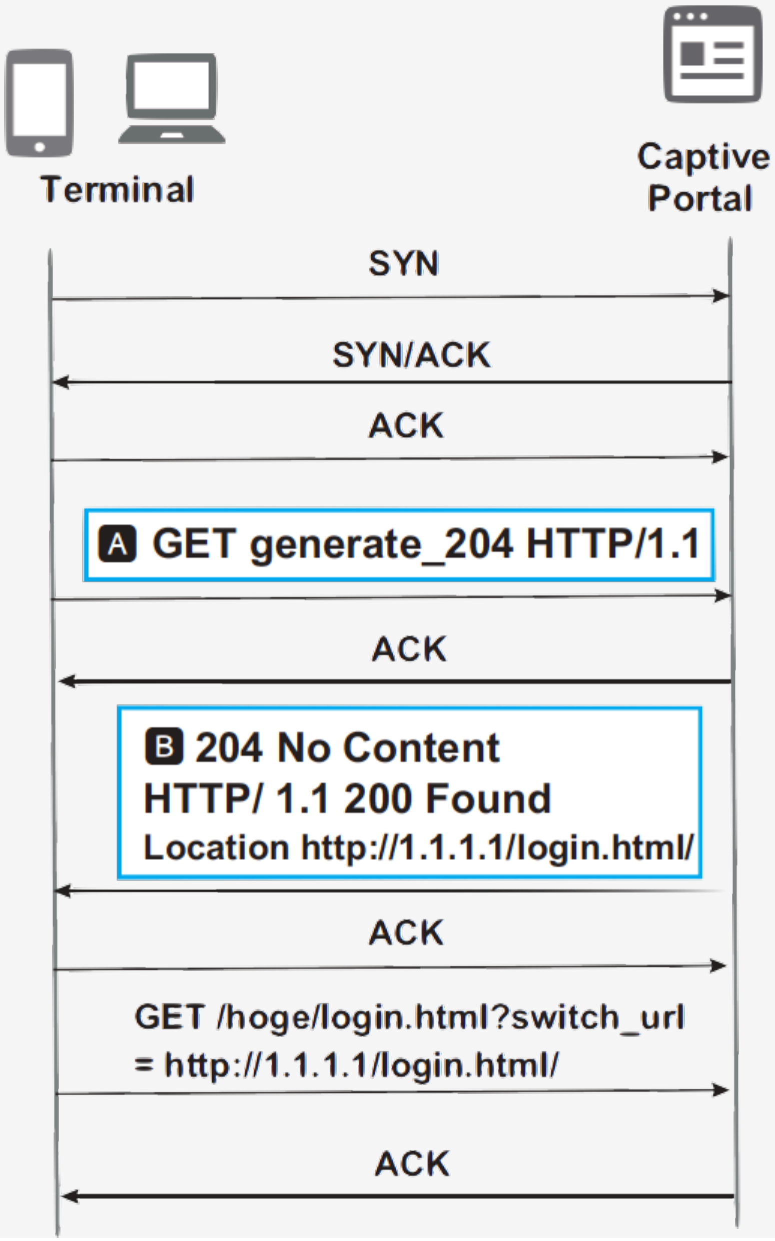
- Most of the capport do not do DNS poisoning for its redirection.
- Most of the probes failed when I set Public DNS(8.8.8.8) for my survey tool kit.

Expected behavior of capport:


Terminal
Captive Portal
SYN
SYN/ACK
ACK
A GET HTTP/1.1
ACK
B HTTP/ 1.1 302 Found
Location http://1.1.1.1/login.html/
ACK
GET /hoge/login.html?switch_url = http://1.1.1.1/login.html/
ACK

□ Replies with either 302 or 307 with a redirection url.

□ Most of OS can detect this type of capport.



Terminal

Captive Portal

SYN

SYN/ACK

ACK

A GET generate_204 HTTP/1.1

ACK

B 204 No Content

HTTP/ 1.1 200 Found

Location http://1.1.1.1/login.html/

ACK

GET /hoge/login.html?switch_url = http://1.1.1.1/login.html/

ACK

Undesirable Behavior

□ Some of capport which response 200 defeats the detection.

□ Some of them also reply “204 & No Content” to Android’s Well-Known Web Page(defeat the detection strategy)

=> All of this model of
capport are deployed
and operated by
the same NSP(in JP).

Why Network Service Provider(NSP) try to defeat capport detection?

- Because of complaint for detection from users?

  - Incognito windows have some troubles with login process (e.g. Google login) or API.

- For marketing (business) reason?

  - They want to get the information from browser's cookie?

- Only Japanese NSP defeat for some conservative reason? How about other countries?

Why are Captive Portals deployed?

□ For Authentication, Payment, Information, Advertisement, Notification.

□ What do NSPs want to get from capport?

- E-mail address – for tracking, marketing.
- Open ID – for tracking, marketing.
- Credit card Info. – to take credit, for payment.
- Browser’s cookie – for marketing.
- UA(user agent) – for judging whether the traffic is users’ true traffic or not.

My Proposal

□ Writing “**capport survey I-D**” will be valuable output for WG.

□ **Conduct a further survey in other main cities or countries.**

- Singapore, San Francisco, London, Australia, Seoul, Beijing, Prague, Chicago etc.

□ Implement capport survey app

- Android app?
- Or adding this contents on IETF app?

Discussion

□ capport detection does not work correctly in Japan. => NSPs cannot provide their service.

We need to cooperate with NSPs not only OS vendors. It is important to meet their demand for our capport solution.

□ Any opinion or ideas for my survey proposal? Any ideal survey items which have to be included for the further survey?

References

- Basic Strategies by Tanaza

  https://success.tanaza.com/s/article/ka0570000004OtgAAE/How-Automatic-Detection-of-Captive-Portal-works

- Basic Strategies by socifi

  https://socifi-doc.atlassian.net/wiki/display/SC/Operating+System+and+Browser+Capabilities+and+Behaviours

- Windows’ strategy

  https://technet.microsoft.com/en-us/library/bc3bf74c-9b46-4258-9d3e-3ed159199df8

  https://msdn.microsoft.com/windows/hardware/drivers/mobilebroadband/captive-portals

- Android’s strategy

  ttps://android.googlesource.com/platform/frameworks/base/+/master/services/core/java/com/android/server/connectivity/NetworkMonitor.java