Melting Pot of Origins

Compromising the Intermediary Web Services that Rehost Websites

Watanabe, T., Shioji, E., Akiyama, M., & Mori, T.

CS 568, Fall 2020

Presenter: Ashesh Singh

1. Motivation

1.1 Motivation: What is this paper about?

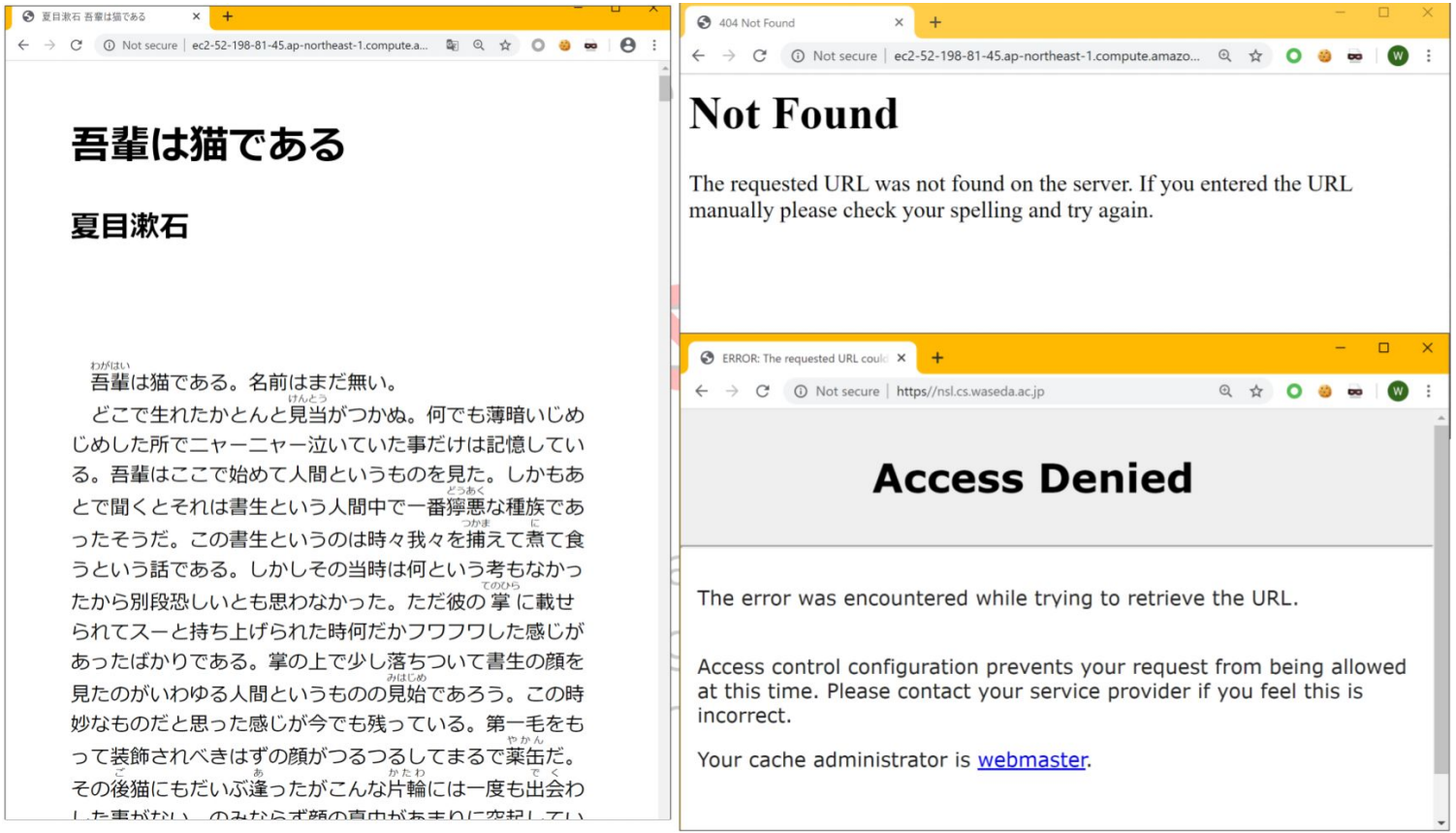
- Common Security flaws in **“web rehosting”** services
- Their possible countermeasures

1.1.1 Motivation: What is “web rehosting”?

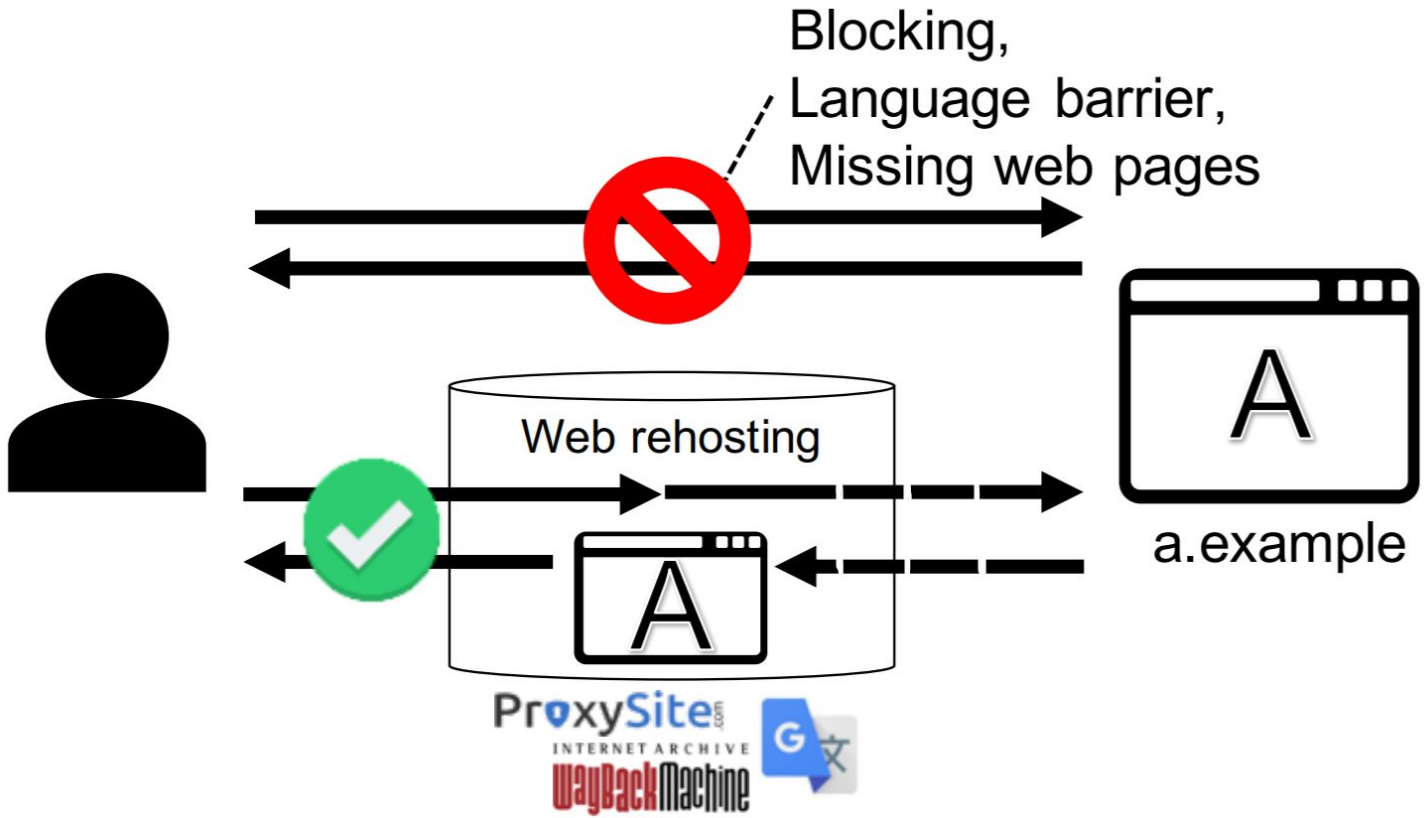
Intermediary web services that aim to remove intrinsic obstacles to web access.

Obstacles:

- Access Blocking
- Language Barriers
- Missing Pages



Obstacles to web access: left ro right; language barrier, missing web page, access blocking.



Web Rehosting

1.1.2 Motivation: Three Problems, 3 Solutions

Access Blocking	Web Proxy (ProxySite, Hide My Ass!, Hide me, Sitenable Proxy, FilterBypass, ProxFree, toolur, hidester, GenMirror, UnblockVideos, Service-α)
Language Barriers	Web Translator (Google Translate, Bing Translator, Weblio, PROMT, Yandex.Translate, Baidu Translate, Service-β)
Missing Pages	Web Archive (Wayback Machine, Google Cache, FreezePage)

1.2 Motivation: Why bother investigating?

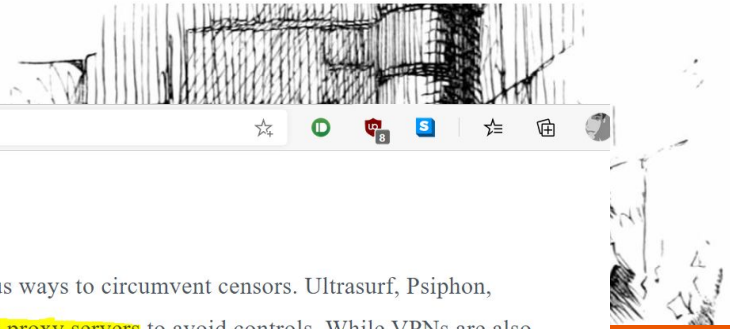
- Intrinsic vulnerability
- Increasing popularity

Hawaii GenTech

Services Archives Categories Contact

Why I Link to WayBackMachine Instead of Original Site

2020-09-07 · Commentary · Web Design



CONTENTS

- Linking to an archive is probably more authoritative than linking to unstable dynamic web content
- Example:
  - Article Content Before
  - Article Content

https://www.cfr.org/backgrounder/media-censorship-china

g the Censors

atic control of news, the Chinese public has found numerous ways to circumvent censors. Ultrasurf, Psiphon, opular software programs that allow Chinese users to set up proxy servers to avoid controls. While VPNs are also ment crackdown on the systems have led users to devise other methods, including the insertion of new IP addresses into ee software program for anonymity—or SSH tunnels, which route all internet traffic through a remote server. ress, between 1 and 8 percent [PDF] of Chinese internet users use proxy servers and VPNs to get around firewalls

> 200 Million session/day combined

2. Attack Surface

2.1 Attack Surface: Web Rehosting Usage

- Direct link (with parameters to target site). Example:
  - https://www.dw.com/de/german-news-service/s-101393
  - https://translate.google.com/translate?hl=en&sl=auto&tl=en&u=https%3A%2F%2Fwww.dw.com%2Fde%2Fgerman-news-service%2Fs-101393

  OR

- Link Input box with page rendered in a iframe/container. *(demos)*

2.2 Attack Surface: Typical service processes

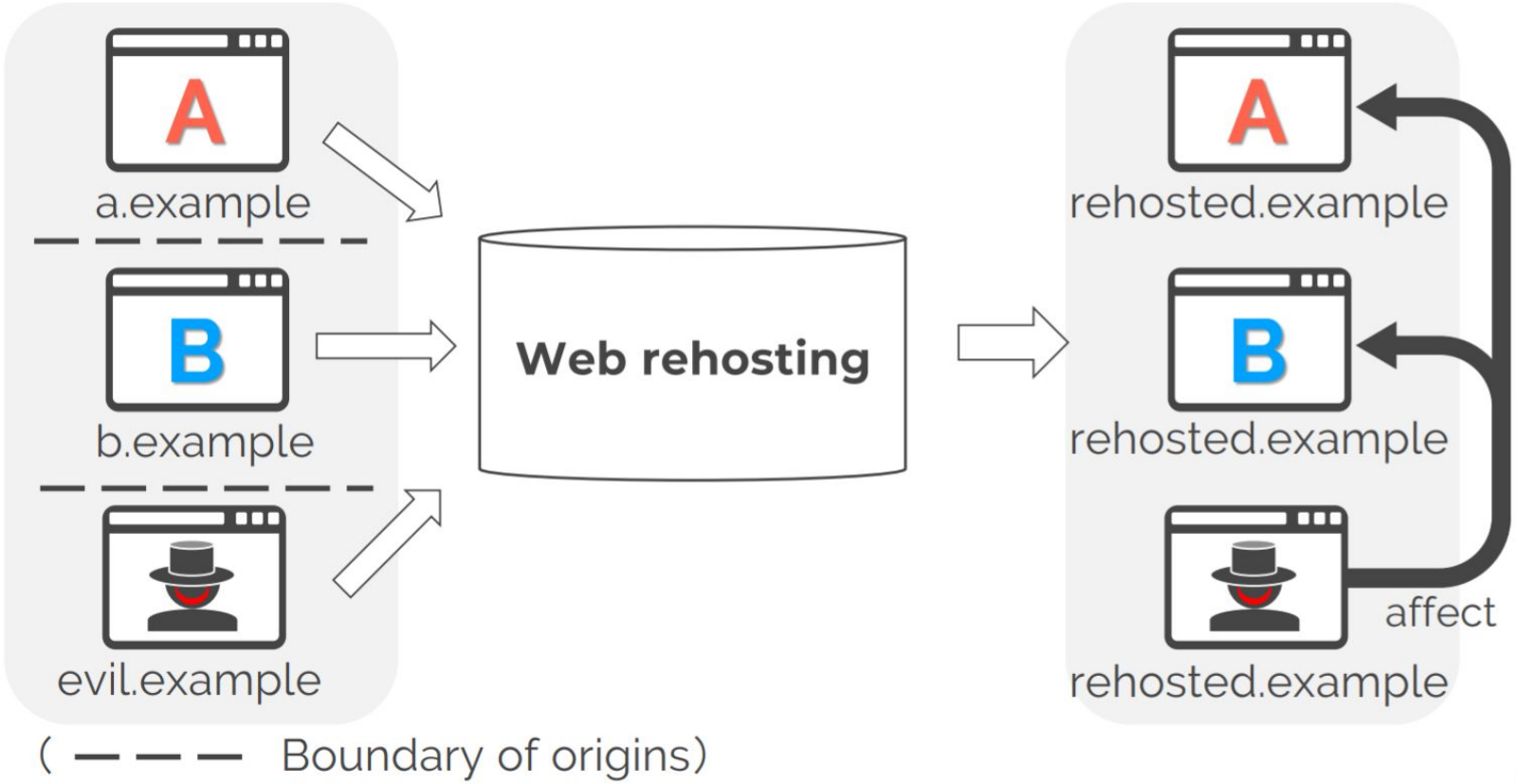
- URL Rewriting (we saw this in previous slide)
- Rehostable File Type
  - Handling Browser Resources
  - JavaScript (in most cases)
- Handling Browser Resources
  - remain resource accesses via JavaScript
  - relay HTTP cookie (web proxy)

2.2 Attack Surface: Bye Bye SOP

- The same-origin policy helps isolate potentially malicious documents, reducing possible attack vectors.

URL	Outcome	Reason
`http://store.company.com/dir2/other.html`	Same origin	Only the path differs
`http://store.company.com/dir/inner/another.html`	Same origin	Only the path differs
`https://store.company.com/page.html`	Failure	Different protocol
`http://store.company.com:81/dir/page.html`	Failure	Different port (`http://` is port 80 by default)
`http://news.company.com/dir/page.html`	Failure	Different host

Single domain name provided is used to access multiple rehosted websites.
SOP is bypassed.



Melting pot of Origins

2.2 Attack Surface: Types

1. Persistent MITM
2. Privilege Abuse
3. Credential Theft
4. History Theft
5. Session Hijacking and Injection

3. Privilege Abuse

3.1 Privilege Abuse

- User grant permission at rehosted benign pages
  Example: https://my-location.org/
- Permission is reused by rehosted malicious page
  Example:
  https://asing80.people.uic.edu/cs568/presentation/demo/2-privilege-abuse.html

  Demo on: https://www.proxysite.com/

3.2 Credential Theft

- User logs in to rehosted benign page and saves credential in password manager
  Example: https://genuine-kepler-e3452c.netlify.app
- Password manager auto-fills credential on fake form of rehosted malicious page
  Example:
  https://asing80.people.uic.edu/cs568/presentation/demo/3-credential-theft.html

  Demo on: https://www.proxysite.com/

3.3 History Theft

- User visits a set rehosted page
  Example: Amazon, GitHub, WSJ
- These sites store some cookies which might be well known (~40% are)
  Example: “_gh_sess”
- Rehosted malicious page retrieves cookie and estimates visited pages.
  Example:
  https://asing80.people.uic.edu/cs568/presentation/demo/4-history-theft.html

  Demo on: https://www.proxysite.com/

3.4 Session Hijacking/Injection

- User visits a set rehosted malicious page
  Example:
  https://asing80.people.uic.edu/cs568/presentation/demo/5-session-hijacking.html
- Malicious page write cookies that has same key as other website
- User visits the target rehosted page under session dictated by the malicious page.
  Example: https://people.uic.edu/cgi-bin/account.cgi

  Demo on: https://www.proxysite.com/

3.5 Persistent MITM

- Exploits ServiceWorkers/AppCache
- Scripts/manifest (by the malicious rehosted page) hosted such that it becomes active on target website.
- User visits target website and all their data is processed through the Service worker or they see different content.

Category	Rehosting Service	Scheme	At least one Vulnerability	Persistent MITM		Privilege Abuse	Credential Theft	History Theft	Session Hijacking & Injection
				SW	AppCache				
Proxy	ProxySite	HTTPS	●	●	●	●	●	●	●
	Hide My Ass!	HTTPS	●	●	●	●	●	●	○
	Hide me	HTTPS	●	●	●	●	●	●	●
	Sitenable Web Proxy	HTTPS	●	●	●	●	●	●	●
	FilterBypass	HTTPS	○	○	●	●	●	●	○
	ProxFree	HTTPS	●	●	●	●	●	●	●
	toolur	HTTPS	●	●	●	●	●	●	●
	hidester	HTTPS	●	●	●	●	●	●	●
	GenMirror	HTTPS	○	○	●	●	●	●	○
	UnblockVideos	HTTPS	●	●	●	●	●	●	●
	Service-α	HTTP/S	●	●	●	●	●	●	●
Translator	Google Translate	HTTPS	●	●	○	○	—	●	—
	Bing Translator	HTTPS	●	○	○	○	—	●	—
	Weblio	HTTPS	●	○	○	●	—	●	—
	PROMT Online	HTTP	●	○	○	○	—	●	—
	Service-β	HTTPS	●	●	○	●	—	●	—
	Yandex.Translate	HTTPS	●	●	●	○	—	●	—
	Baidu Translate	HTTP	●	○	○	○	—	●	—
Archive	Wayback Machine	HTTPS	●	○	●	●	—	●	—
	Google Cache	HTTP/S	●	○	○	●	—	●	—
	FreezePage	HTTP	○	○	○	○	—	○	—

18 out of 21

Summary

Mitigation?

1. Separate domain names for each rehosted page https://rehosted.example/?url=a.example https://a-example.rehosted.example/
2. Generate tentative URL inaccessible by 3rd party Inhibit direct links
3. Disable SW and AppCache (attack I)
4. Use HTTPOnly (attack V)

References

- MDN Web Docs
- Melting Pot of Origins: Compromising the Intermediary Web Services that Rehost Websites
- NDSS 2020 Melting Pot of Origins: Compromising the Intermediary Web Services that Rehost Websites