

Talking with Familiar Strangers: An Empirical Study on HTTPS Context Confusion Attacks

Mingming Zhang¹, Xiaofeng Zheng¹, Kaiwen Shen, Ziqiao Kong, Chaoyi Lu,
Yu Wang, Haixin Duan, Shuang Hao, Baojun Liu and Min Yang

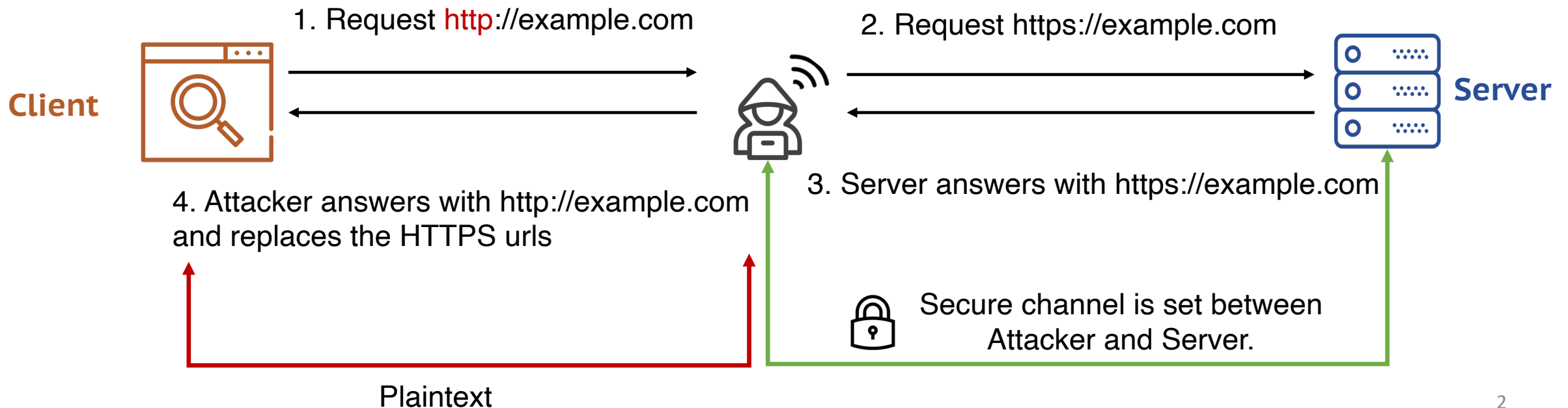
Email: zmm18@mails.tsinghua.edu.cn



HTTPS Man-in-the-middle (MITM) Attacks

- **SSL Strip Attack (Moxie Marlinspike, 2009)**

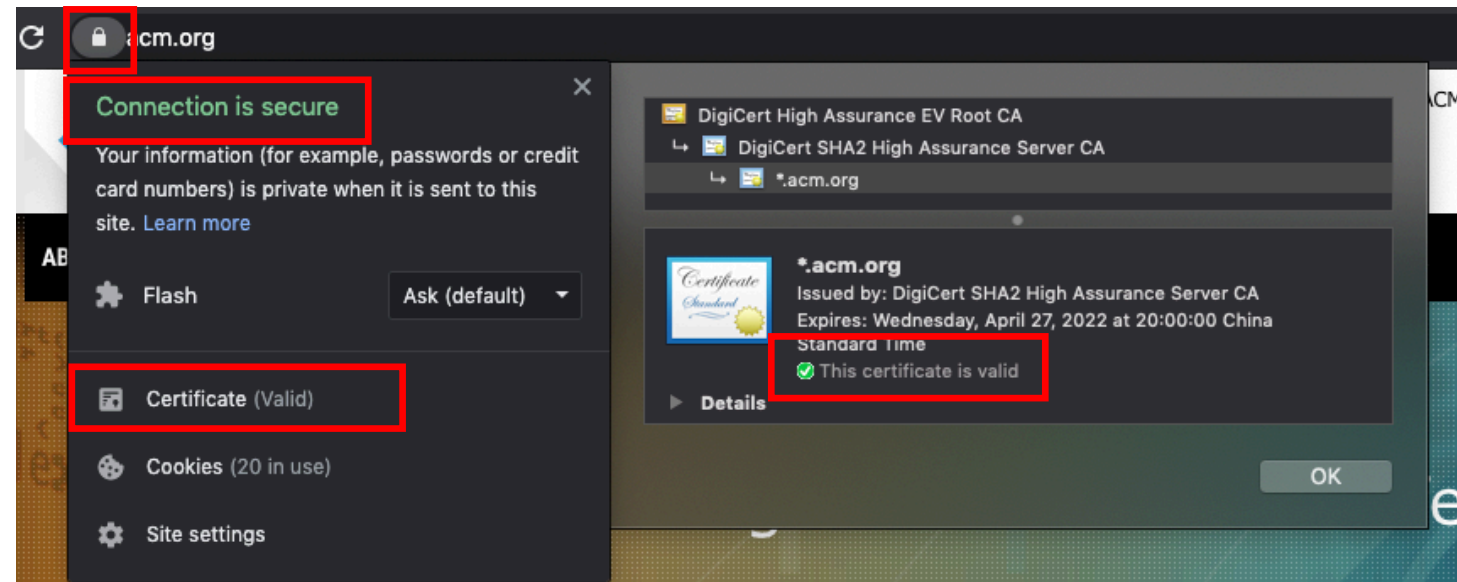
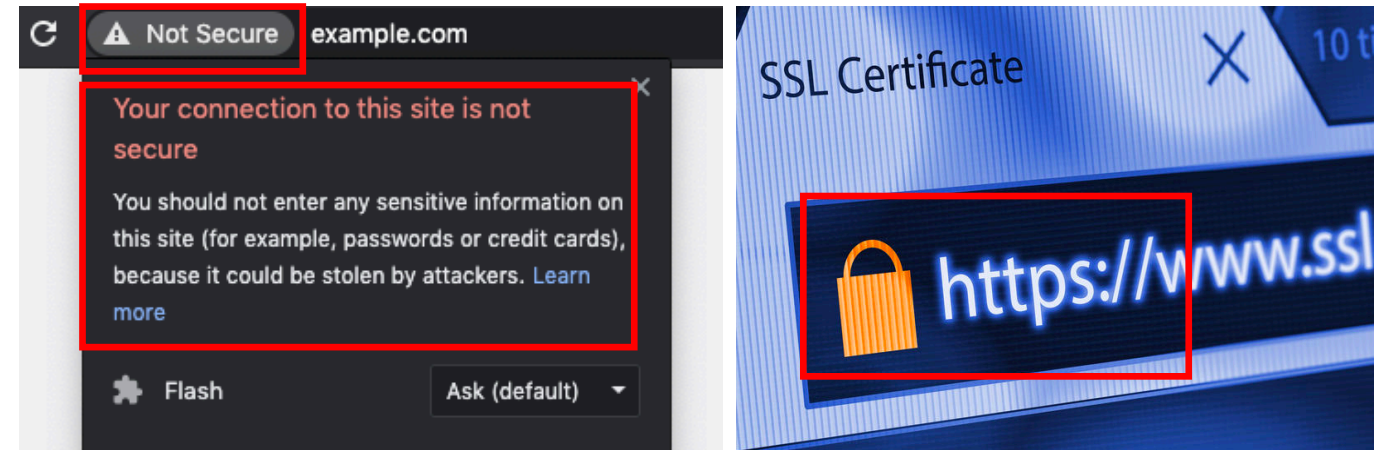
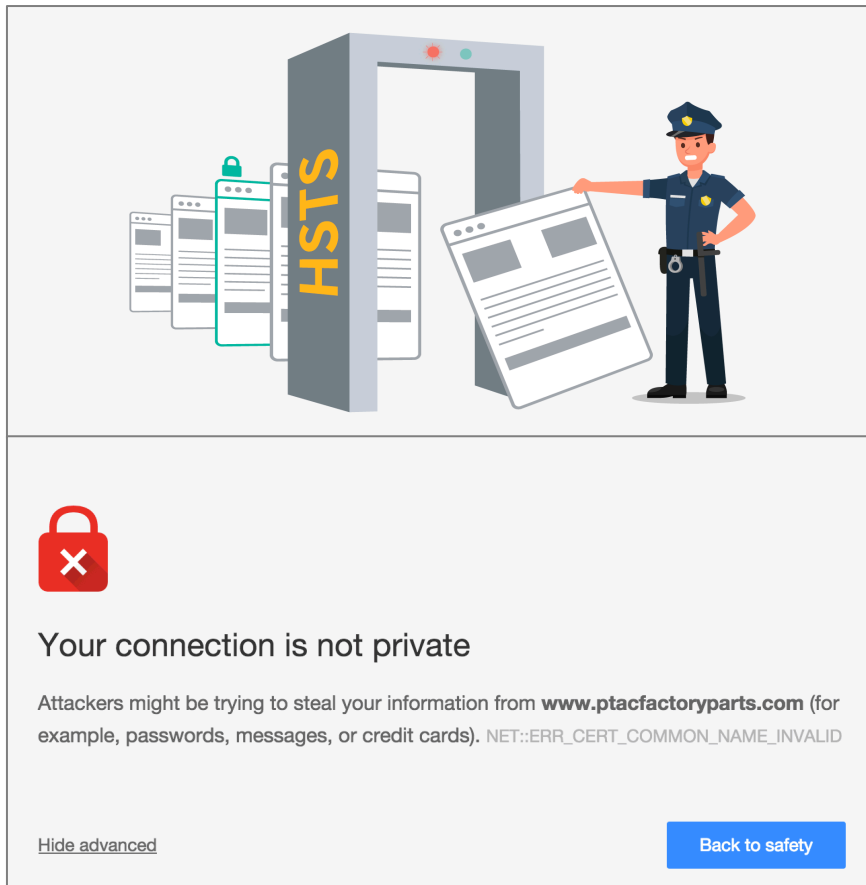
- Intercept the **initial** HTTP connection
- Replace the secure links returned by the server with plaintext ones
- Downgrade the SSL channel



HTTPS Man-in-the-middle (MITM) Attacks

- **Mitigation**

- Enforce HSTS policy
- Browser UI security indicators

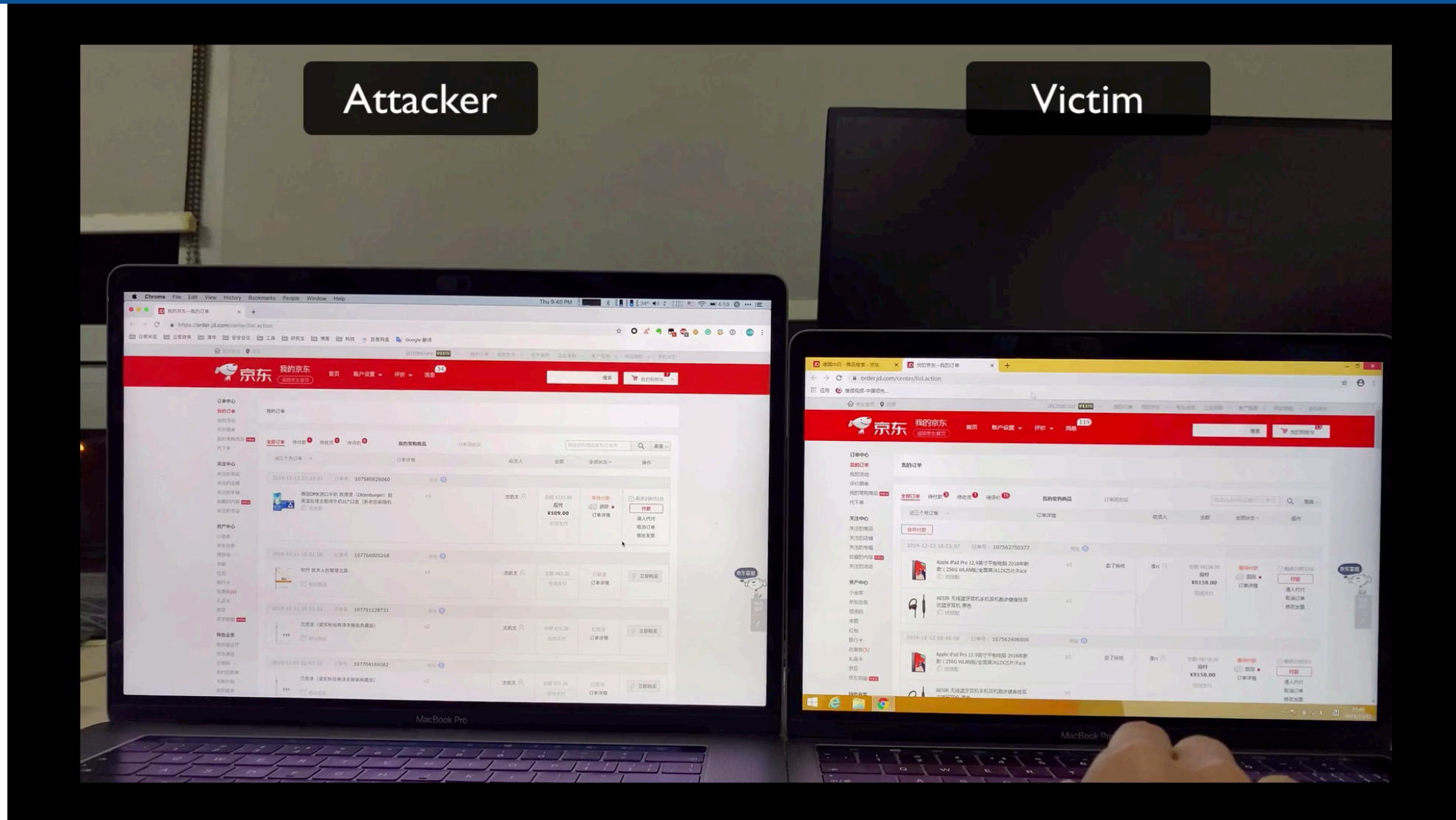


With these **security policies** being well-deployed on one website,

the HTTPS-protected websites are secure enough



Demo: Payment Hijacking on JD Shopping

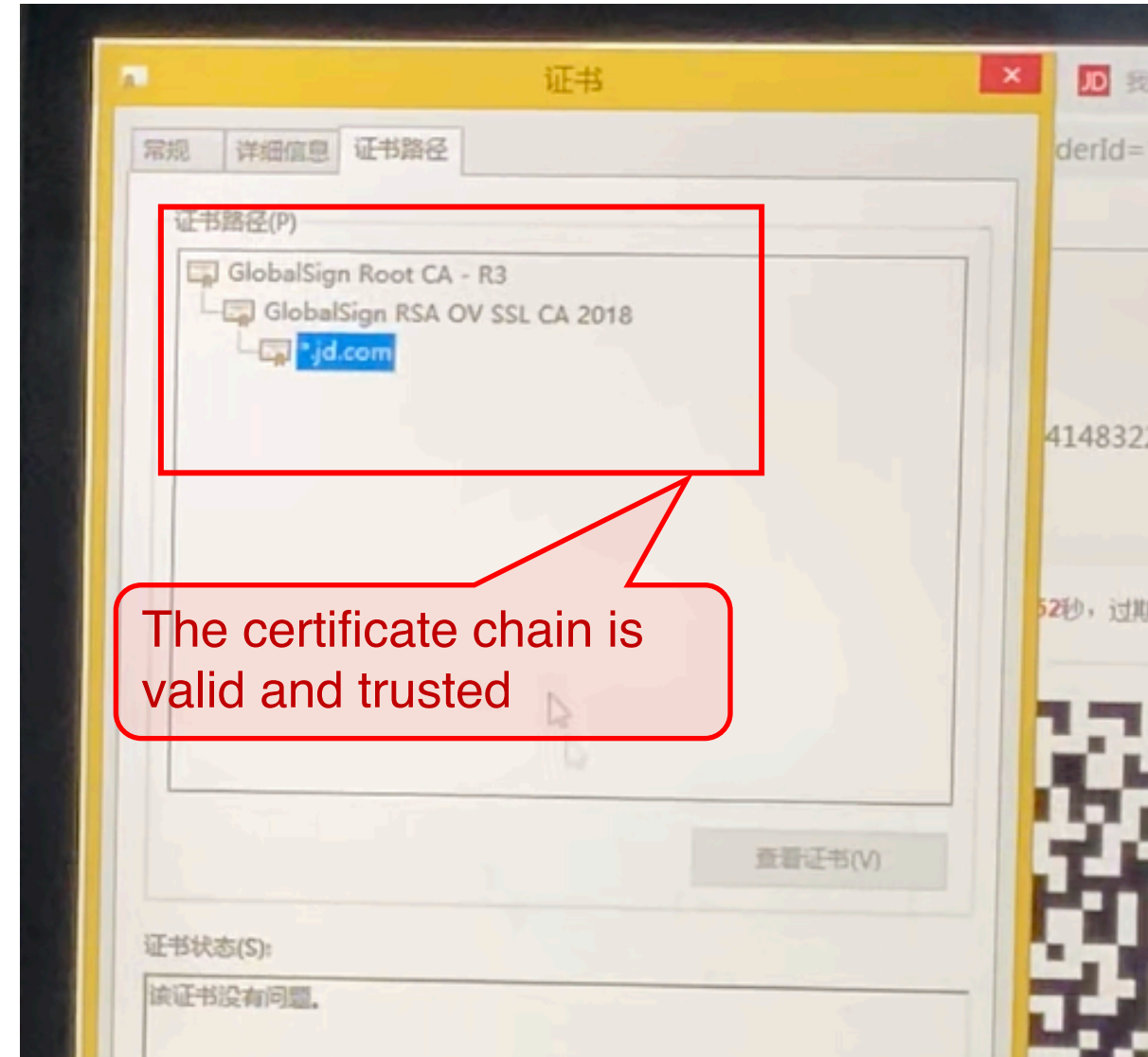


JD Shopping is a large online shopping website in China (Alexa Rank 10)

Demo: Payment Hijacking on JD Shopping

SCC Attack

- **Unnoticeable** to users
- **Undetectable** to browsers



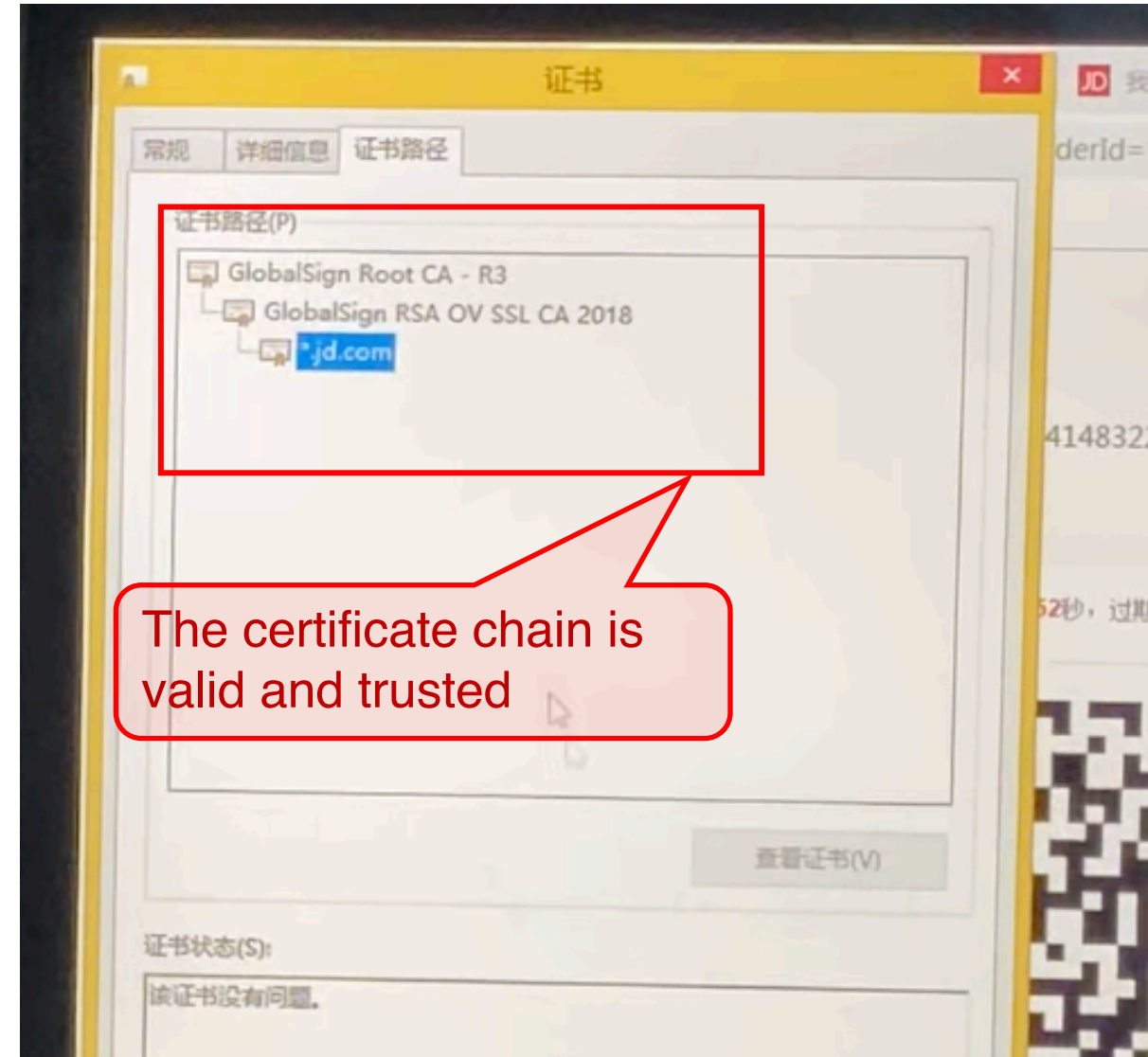
Demo: Payment Hijacking on JD Shopping

SCC Attack

- **Unnoticeable** to users
- **Undetectable** to browsers

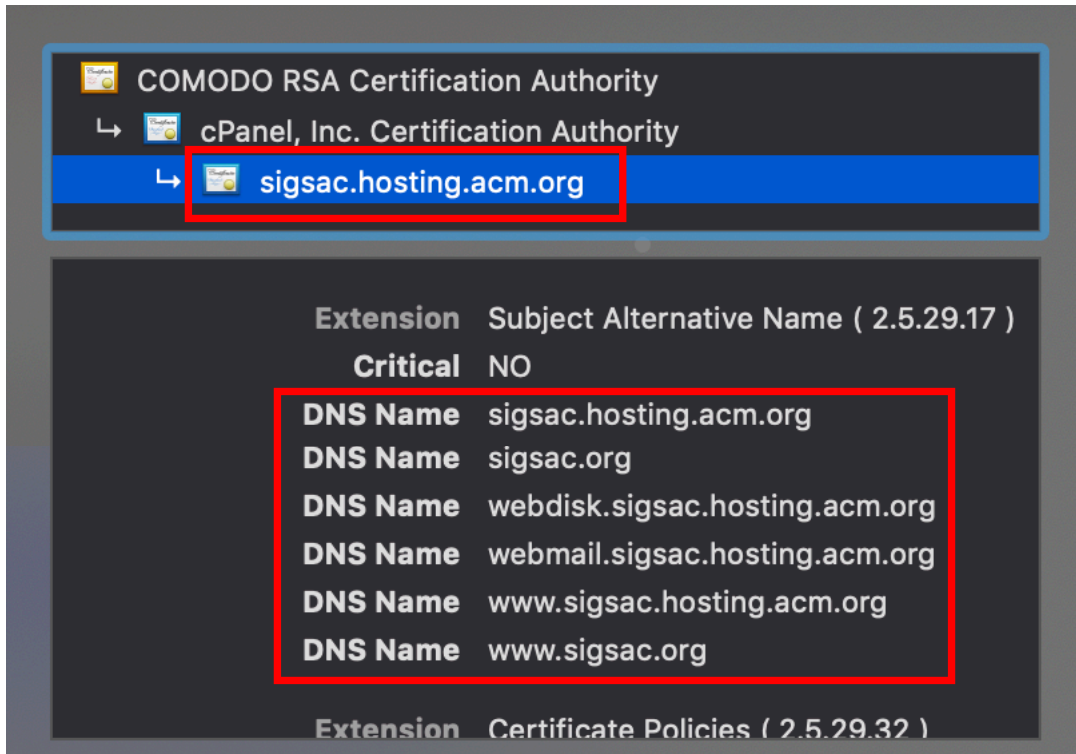


How?



Certificate Sharing

- **One certificate for multiple domains**
 - Multi-domain and Wildcard certificates
- **Multiple servers with one certificate**
 - Sharing the same certificate is common (e.g., CDN nodes, virtual hosts, associated services, commercial cooperation parties)



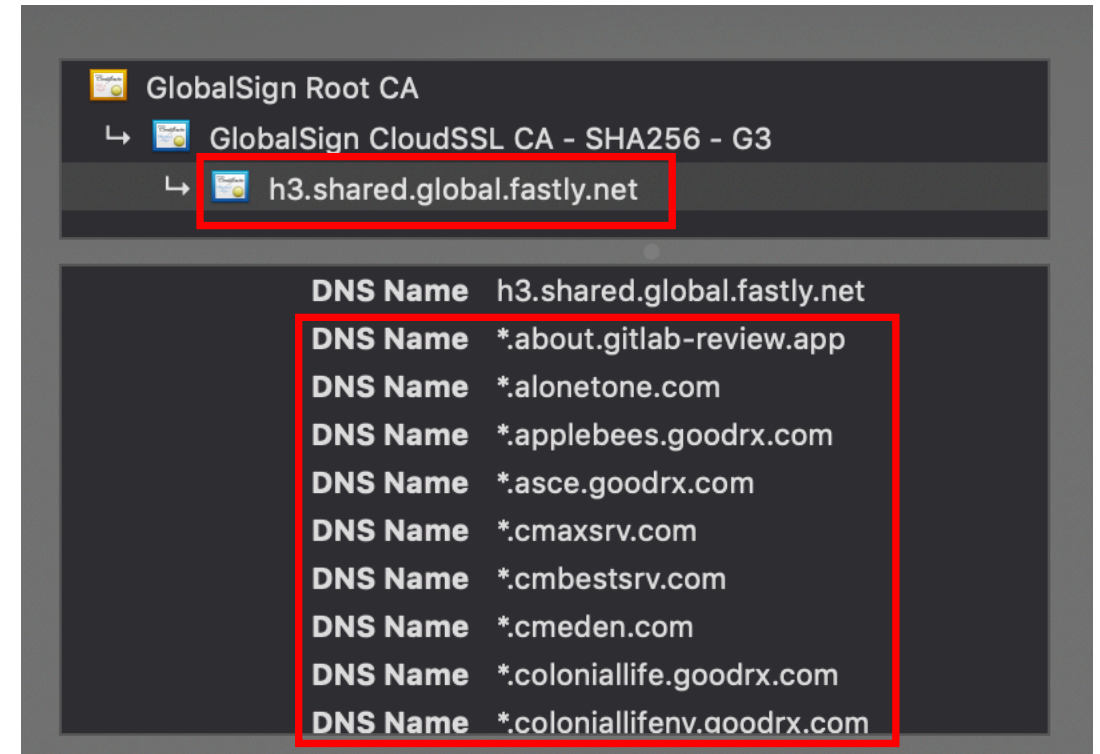
COMODO RSA Certification Authority

↳ cPanel, Inc. Certification Authority

↳ **sigsac.hosting.acm.org**

Extension	Subject Alternative Name (2.5.29.17)
Critical	NO
DNS Name	sigsac.hosting.acm.org
DNS Name	sigsac.org
DNS Name	webdisk.sigsac.hosting.acm.org
DNS Name	webmail.sigsac.hosting.acm.org
DNS Name	www.sigsac.hosting.acm.org
DNS Name	www.sigsac.org

Extension Certificate Policies (2.5.29.32)



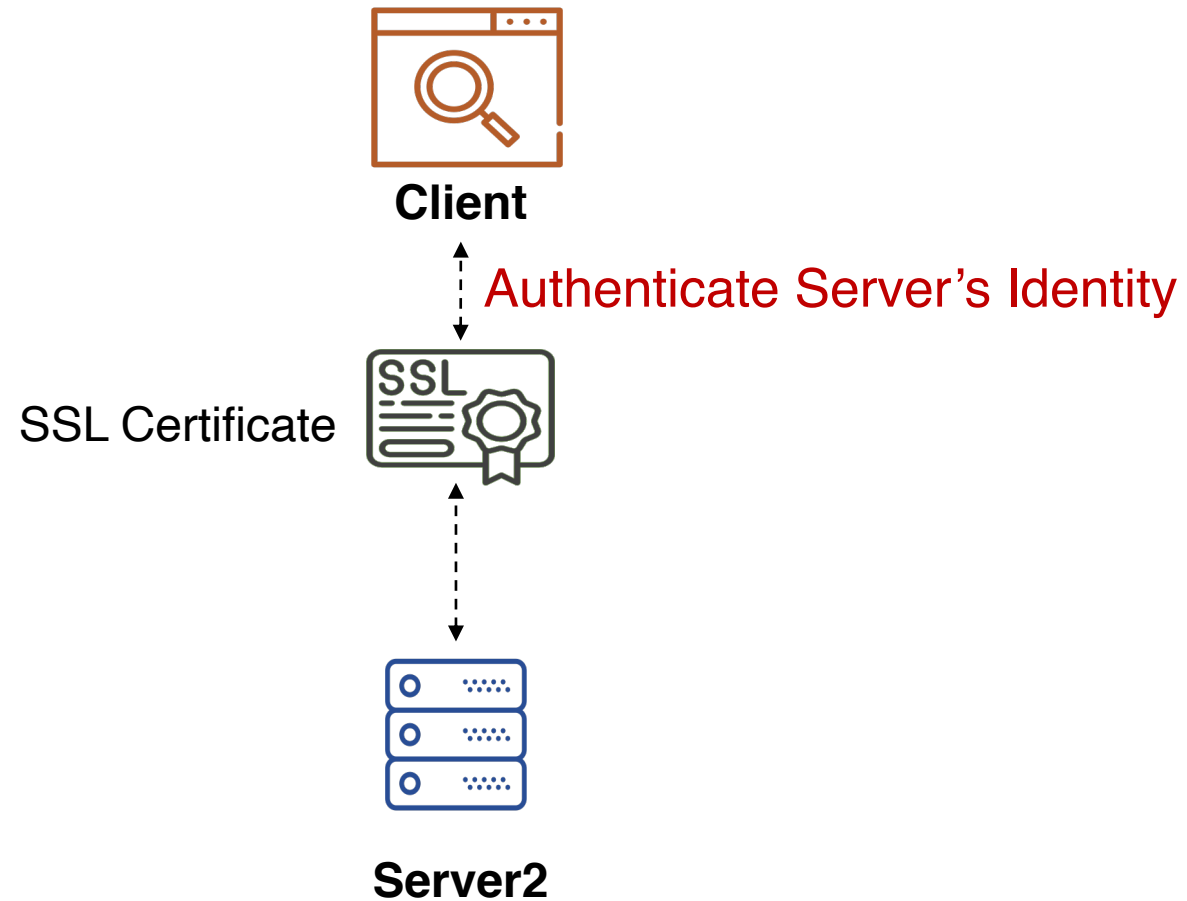
GlobalSign Root CA

↳ GlobalSign CloudSSL CA - SHA256 - G3

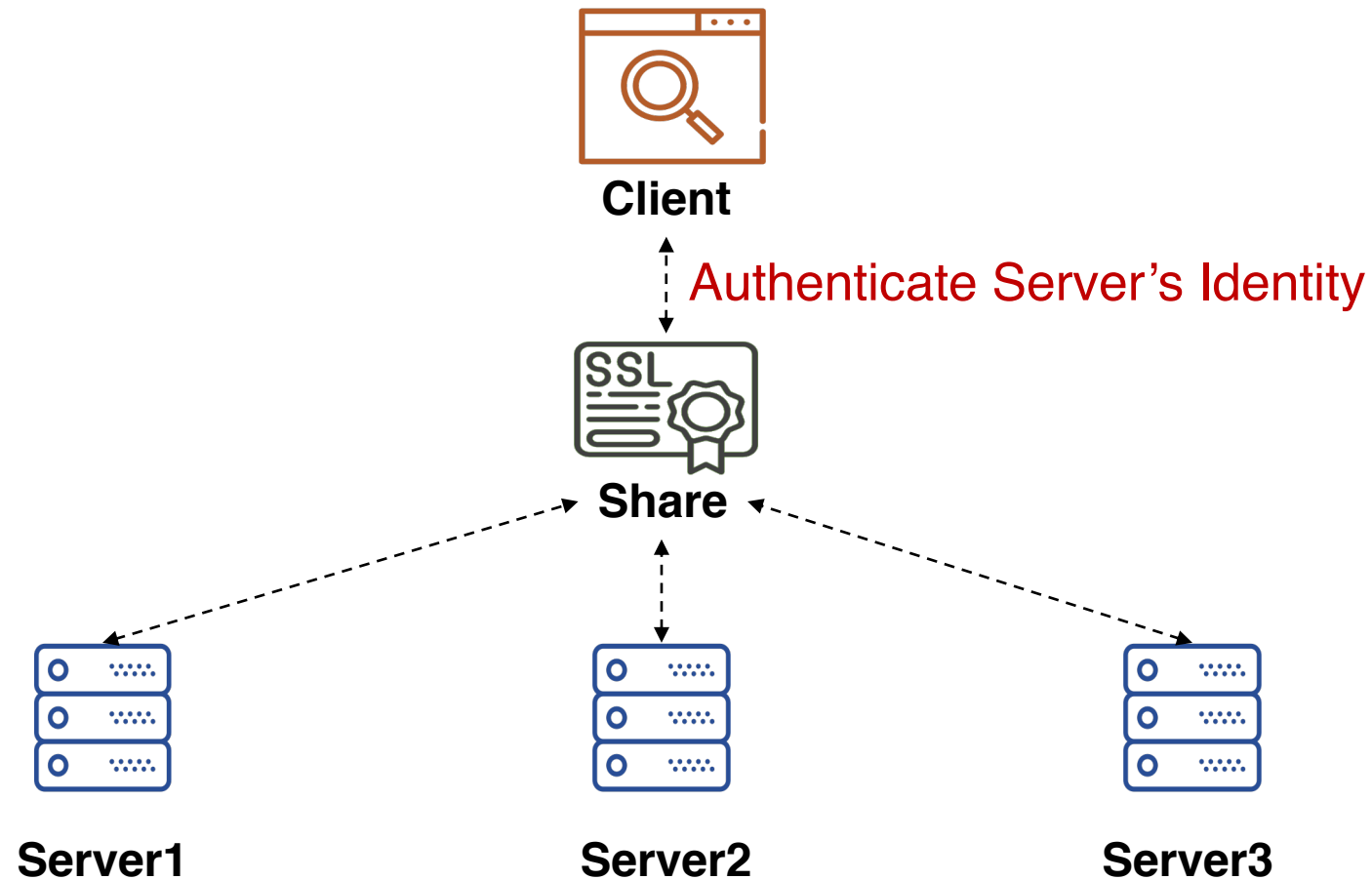
↳ **h3.shared.global.fastly.net**

DNS Name	h3.shared.global.fastly.net
DNS Name	*.about.gitlab-review.app
DNS Name	*.alonetone.com
DNS Name	*.applebees.goodrx.com
DNS Name	*.asce.goodrx.com
DNS Name	*.cmaxsrv.com
DNS Name	*.cmbestsrv.com
DNS Name	*.cmeden.com
DNS Name	*.coloniallife.goodrx.com
DNS Name	*.coloniallifenv.goodrx.com

Certificate Sharing

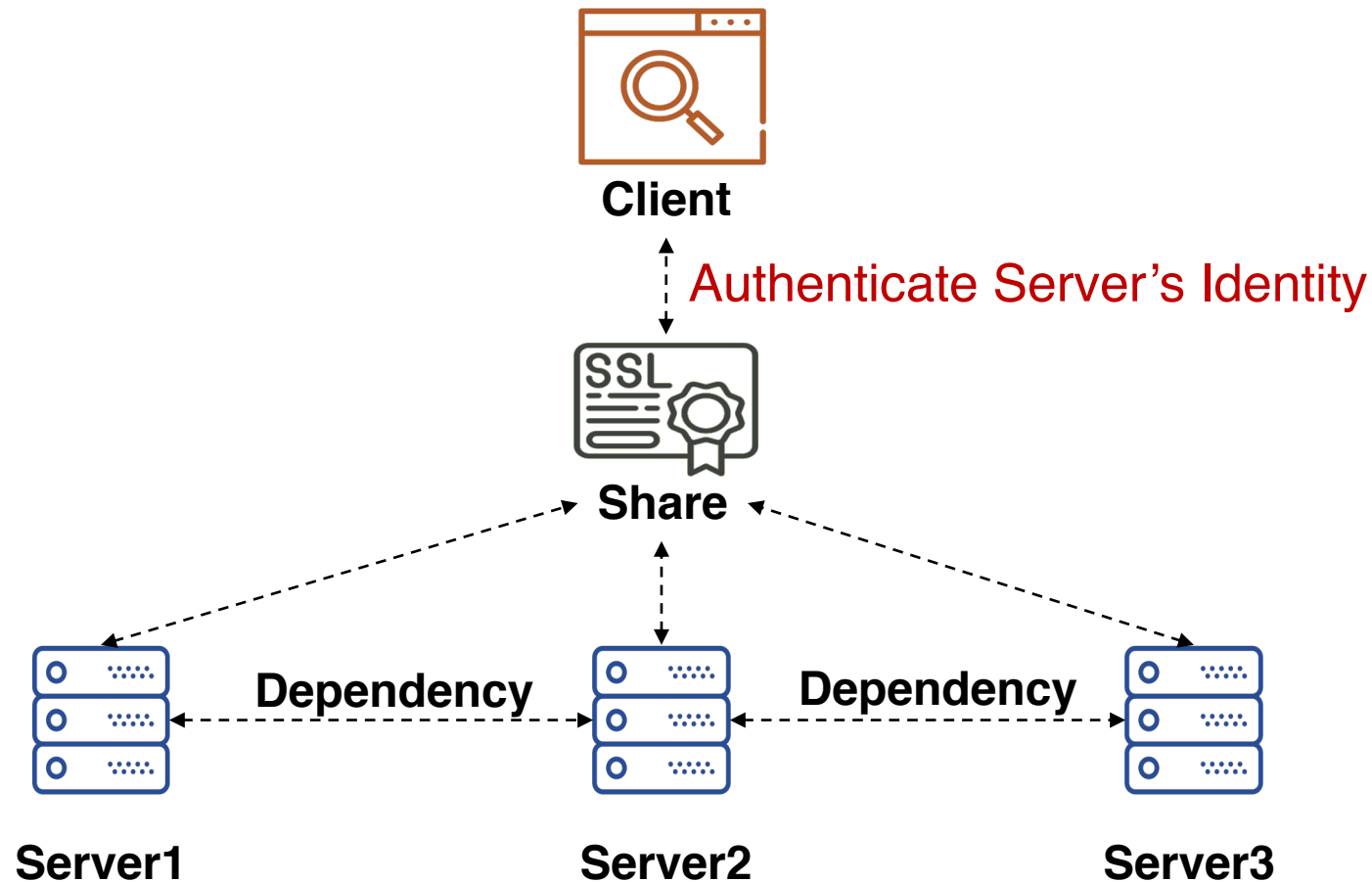


Certificate Sharing



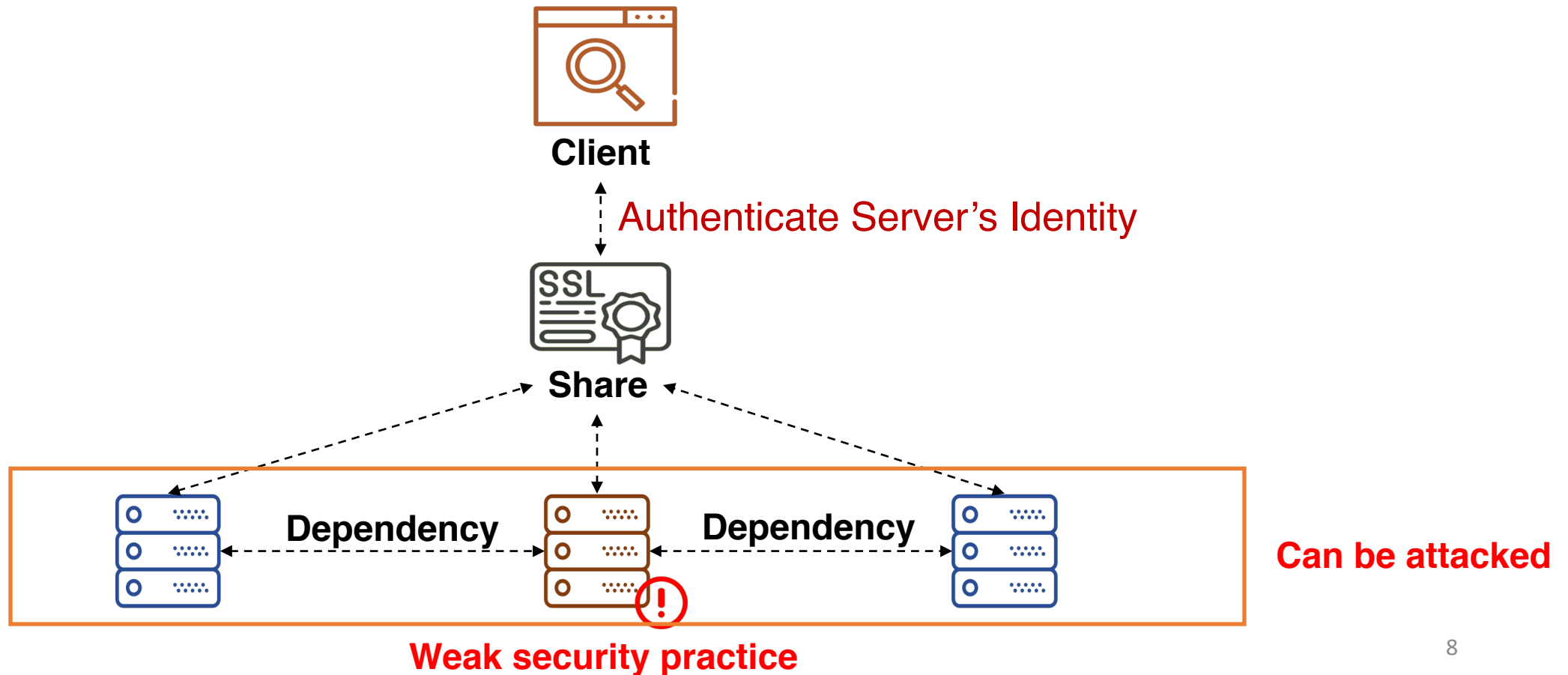
Certificate Sharing

- The shared TLS certificates lead to **security dependencies** among different servers/parties.

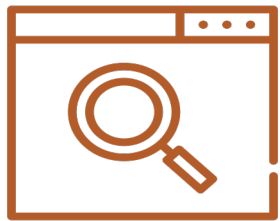


Certificate Sharing

- The shared TLS certificates lead to **security dependencies** among different servers/parties.



Attack Flow



Client



MITM

(a.example.com, a.a.a.a)



Server A



*.example.com

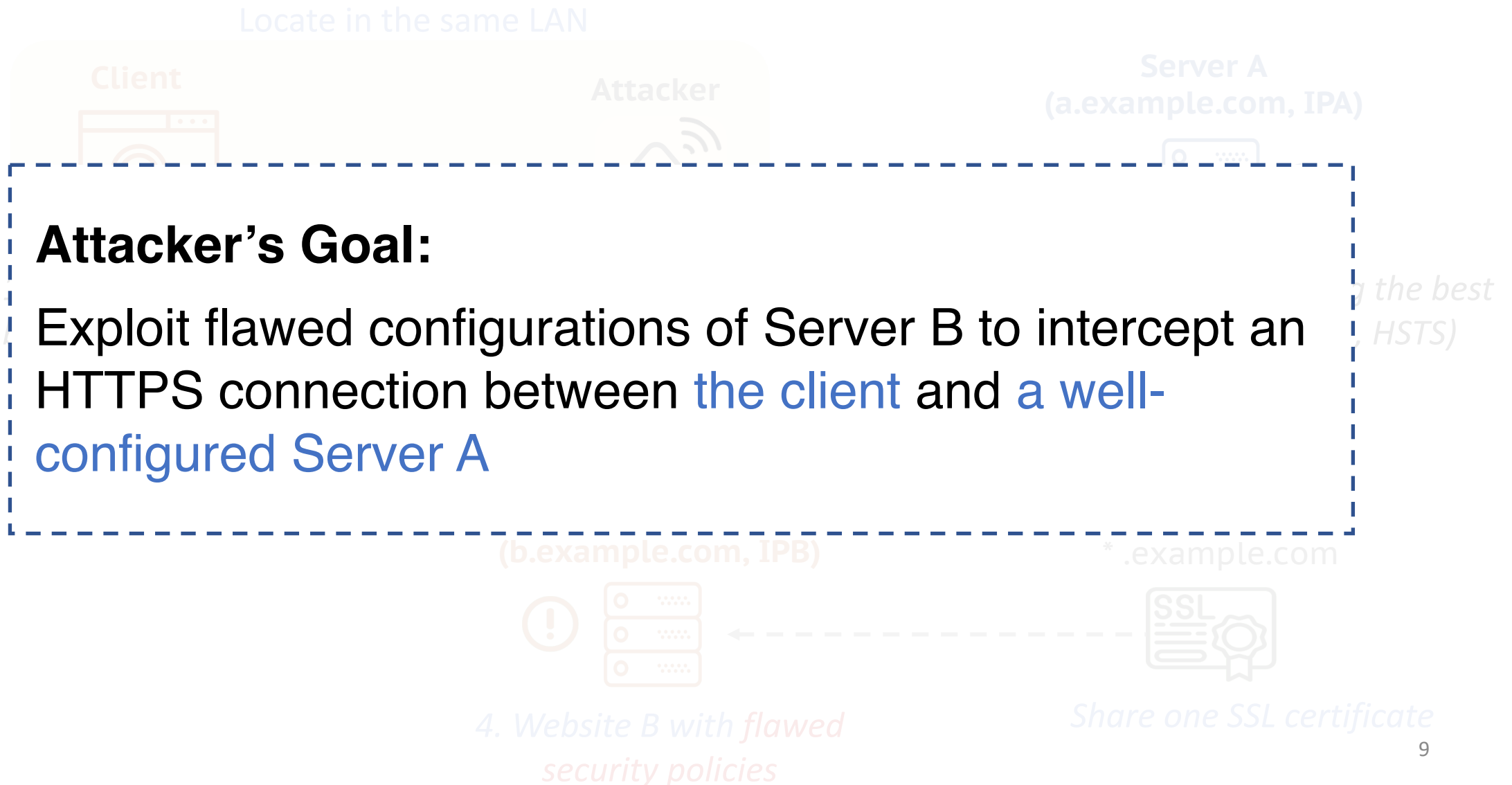


Server B

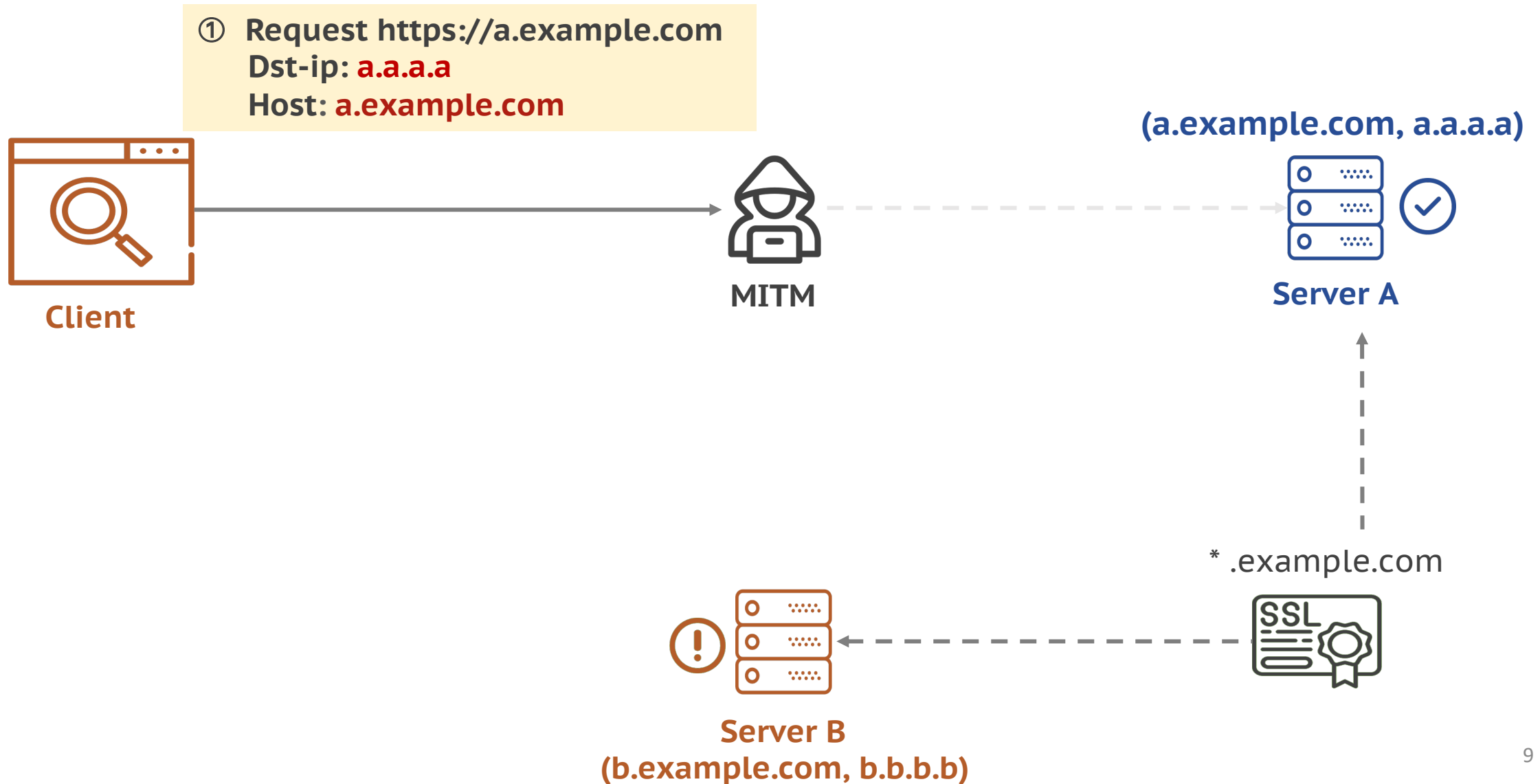
(b.example.com, b.b.b.b)



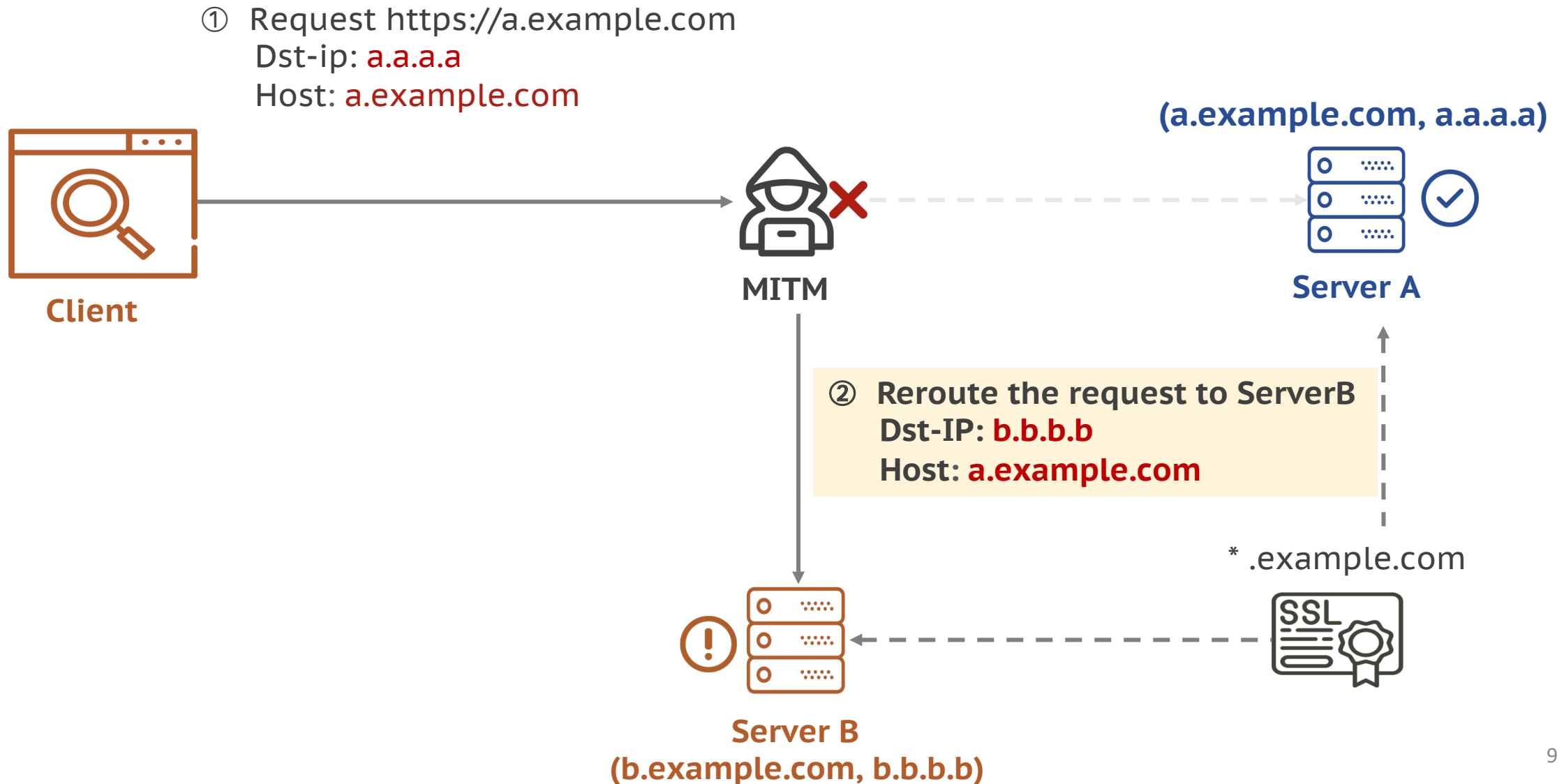
Attack Flow



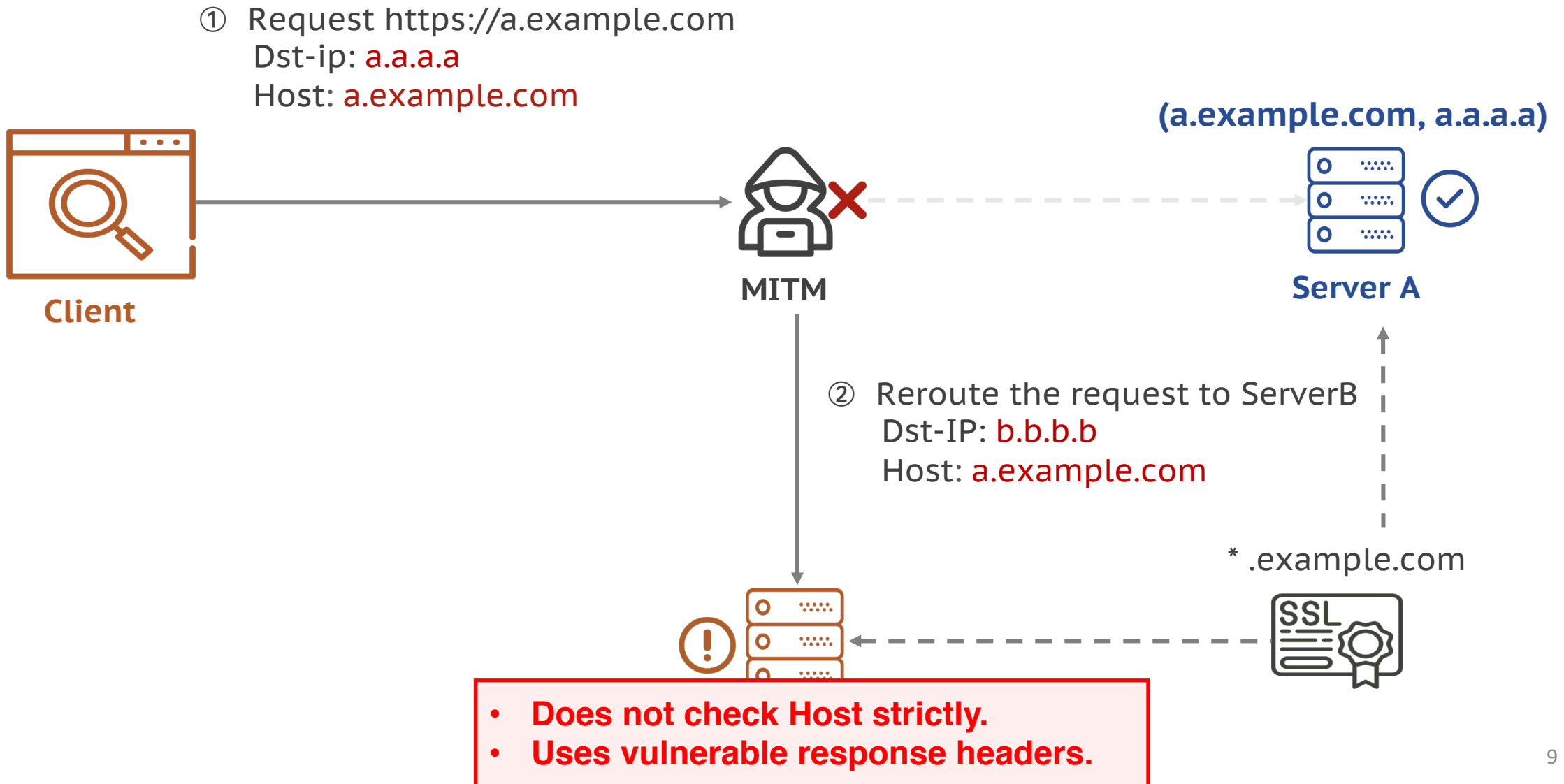
Attack Flow



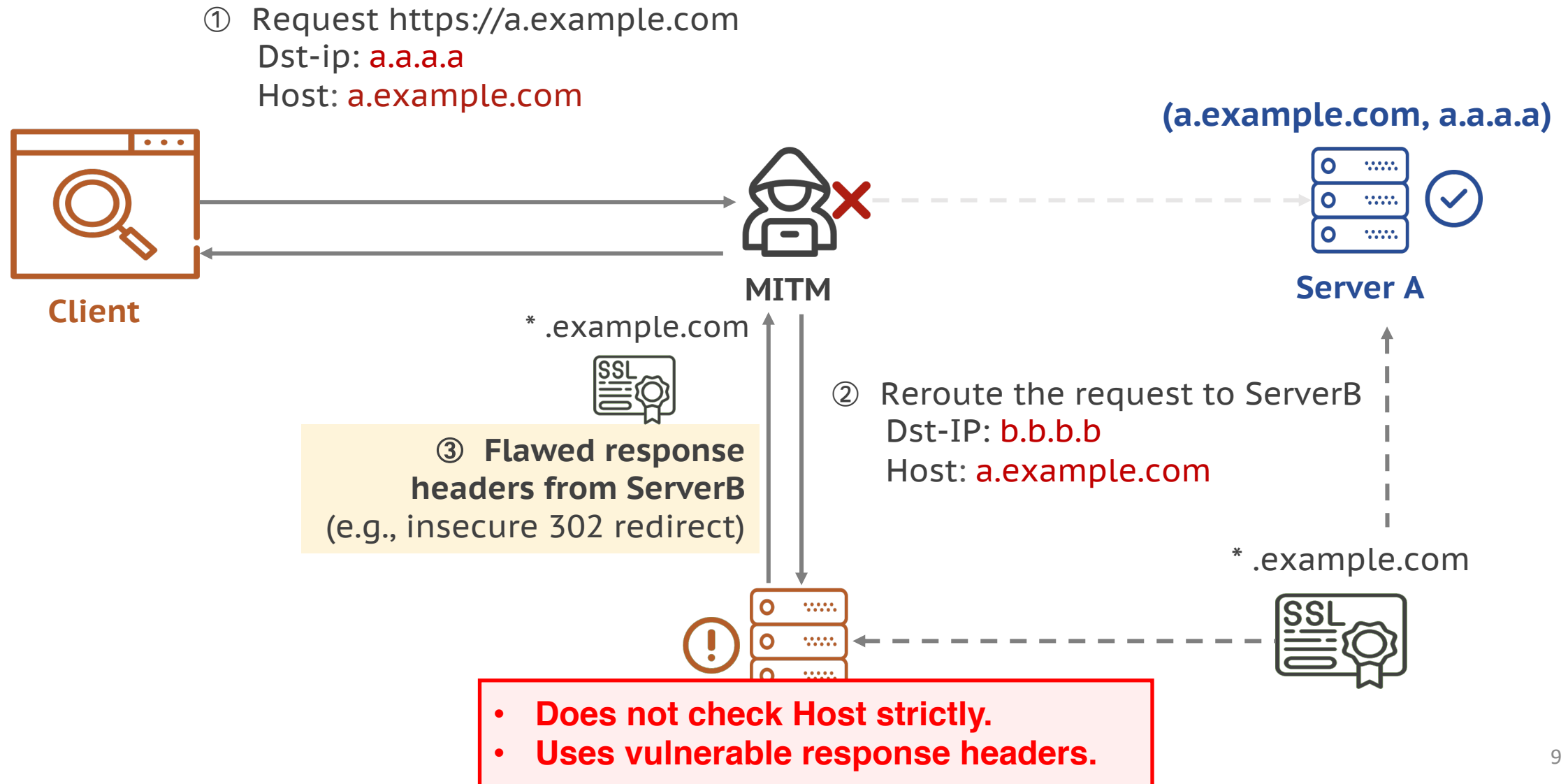
Attack Flow



Attack Flow

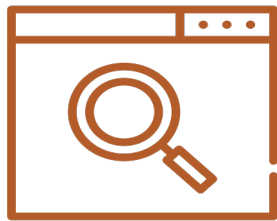


Attack Flow



Attack Flow

- ① Request https://a.example.com
Dst-ip: a.a.a.a
Host: a.example.com



Client



MITM

(a.example.com, a.a.a.a)



Server A

*.example.com



- ③ Flawed response headers from ServerB

- ② Reroute the request to ServerB
Dst-IP: b.b.b.b
Host: a.example.com

*.example.com



- ④ Enforce the **vulnerable policies** of ServerB for ServerA



- Does not check Host strictly.
- Uses vulnerable response headers.

HTTPS Context Confusion Attack (SCC Attack)

HTTPS MITM attacks leveraging shared TLS certificates

- **Goal:** Exploit flawed configurations of Server B to intercept an HTTPS connection between the client and a well-configured Server A.
- **Looking from client-side**
 - Client is actually talking with Server B (not Server A)
 - Can not be detected by browsers
 - Secure browsing context confusion for programs and users

Types of SCC Attack

Downgrade HTTPS to HTTP using the
insecure 3xx redirects from ServerB

HTTPS Downgrading Attack

One-shot Downgrade (Down-1)

Multi-hops Downgrade (Down-2)

SCC Attack

HSTS Bypassing Attack

Clear HSTS Policy (HSTS-1)

Cancel HSTS for Subdomain (HSTS-2)

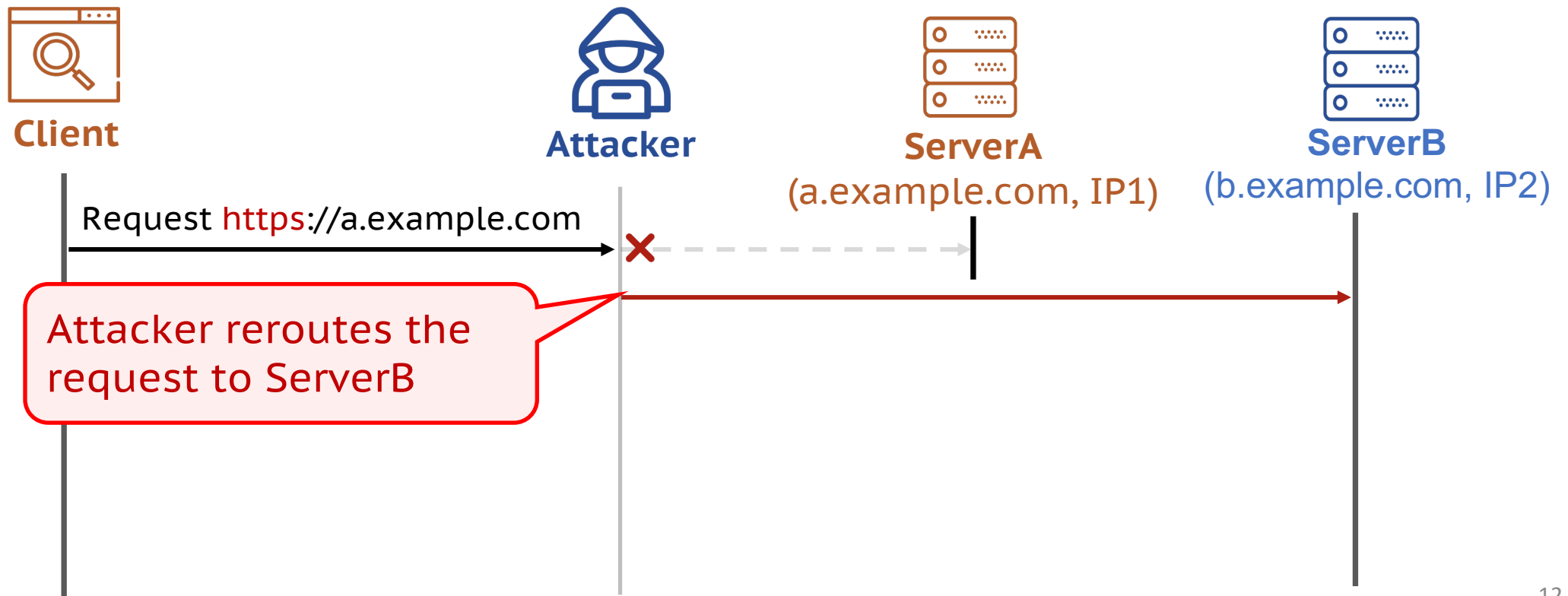
Bypass HSTS Policy using **flawed Strict-Transport-Security (STS) headers** from ServerB.

Decrease HSTS Validity Period (HSTS-3)

SCC Attack: Bypassing HTTPS Security Policies

Type 1: HTTPS Downgrading Attack

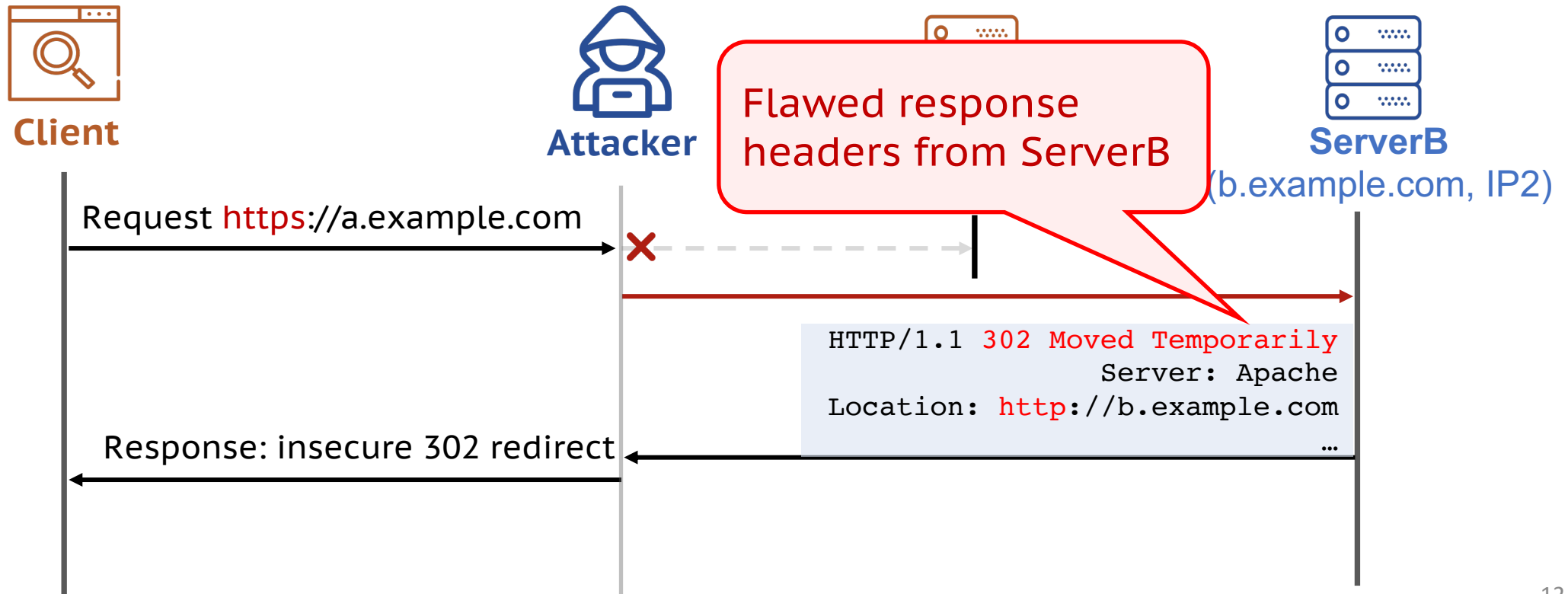
Downgrading HTTPS to HTTP using **insecure 3xx redirects**



SCC Attack: Bypassing HTTPS Security Policies

Type 1: HTTPS Downgrading Attack

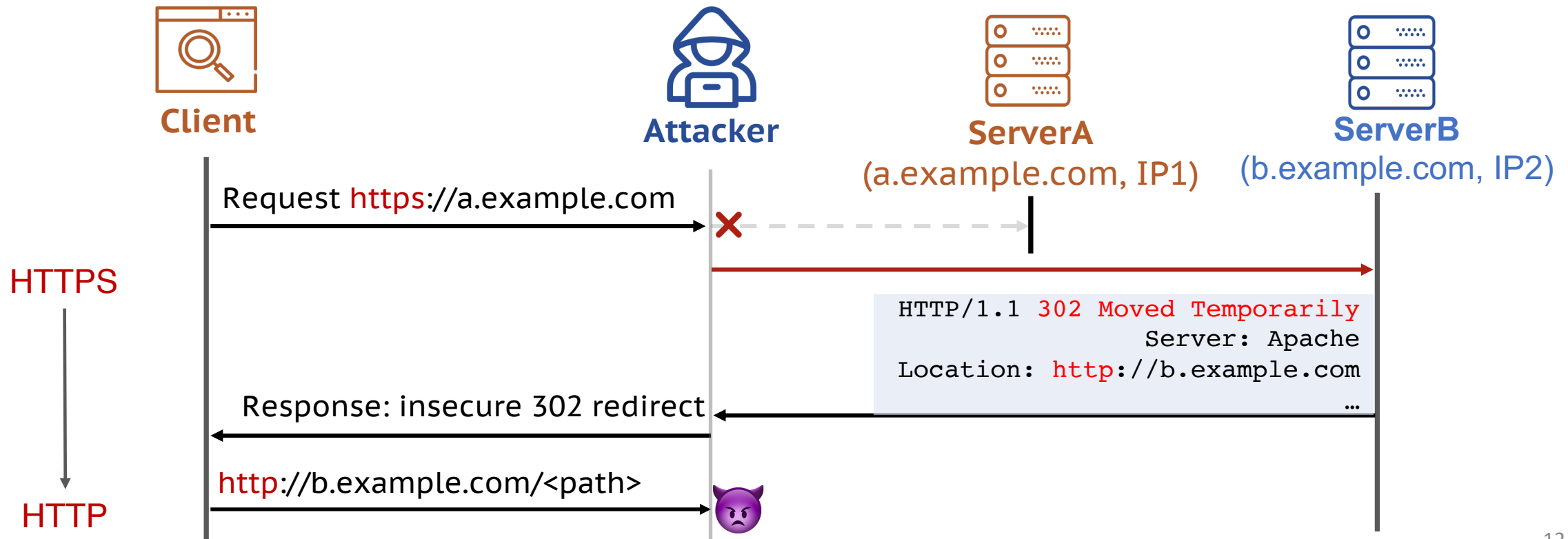
Downgrading HTTPS to HTTP using **insecure 3xx redirects**



SCC Attack: Bypassing HTTPS Security Policies

Type 1: HTTPS Downgrading Attack

Downgrading HTTPS to HTTP using **insecure 3xx redirects**



SCC Attack: Bypassing HTTPS Security Policies

Type 2: HSTS Bypassing Attack

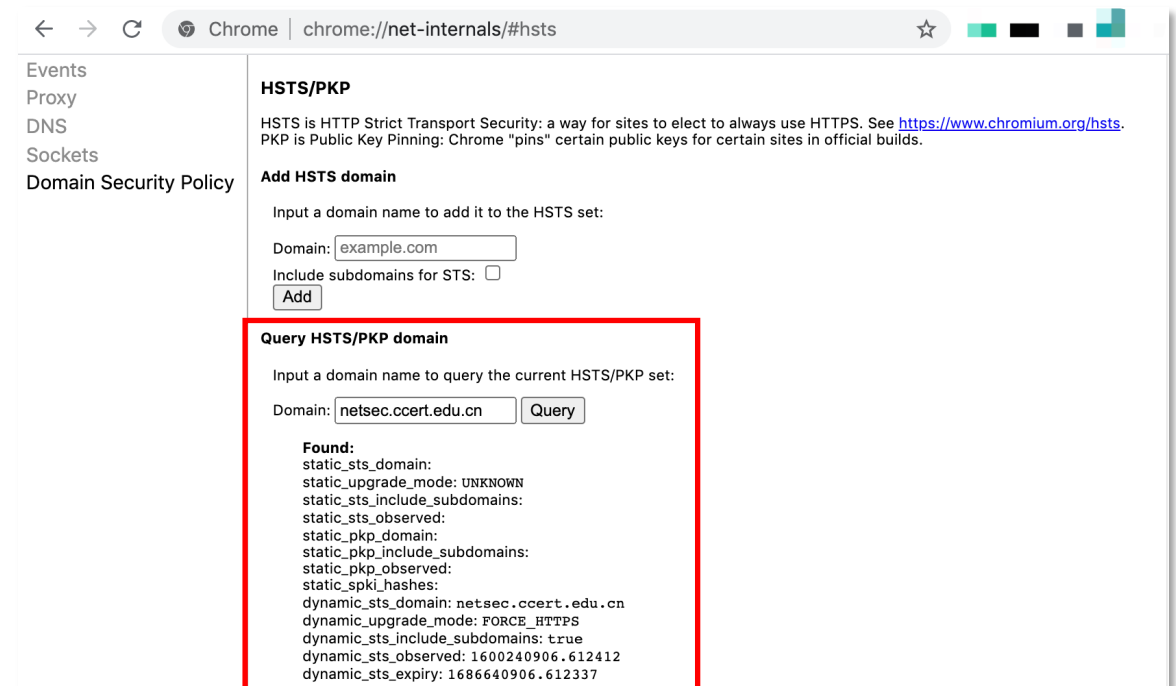
Bypassing HSTS Policy using **flawed Strict-Transport-Security (STS) header**.

Server: specify HSTS Policy by STS Header

Strict-Transport-Security:
max-age=15552000; includeSubDomains; preload



Browser: enforce HSTS Policy for the Server



SCC Attack: Bypassing HTTPS Security Policies

Type 2: HSTS Bypassing Attack

Bypassing HSTS Policy using **flawed Strict-Transport-Security (STS) header**.

Flawed STS Header of ServerB

Strict-Transport-Security: **max-age=0**

Strict-Transport-Security: **<no includeSubdomain>**

Strict-Transport-Security: **max-age=<smaller-than-ServerA>**



Browser Action

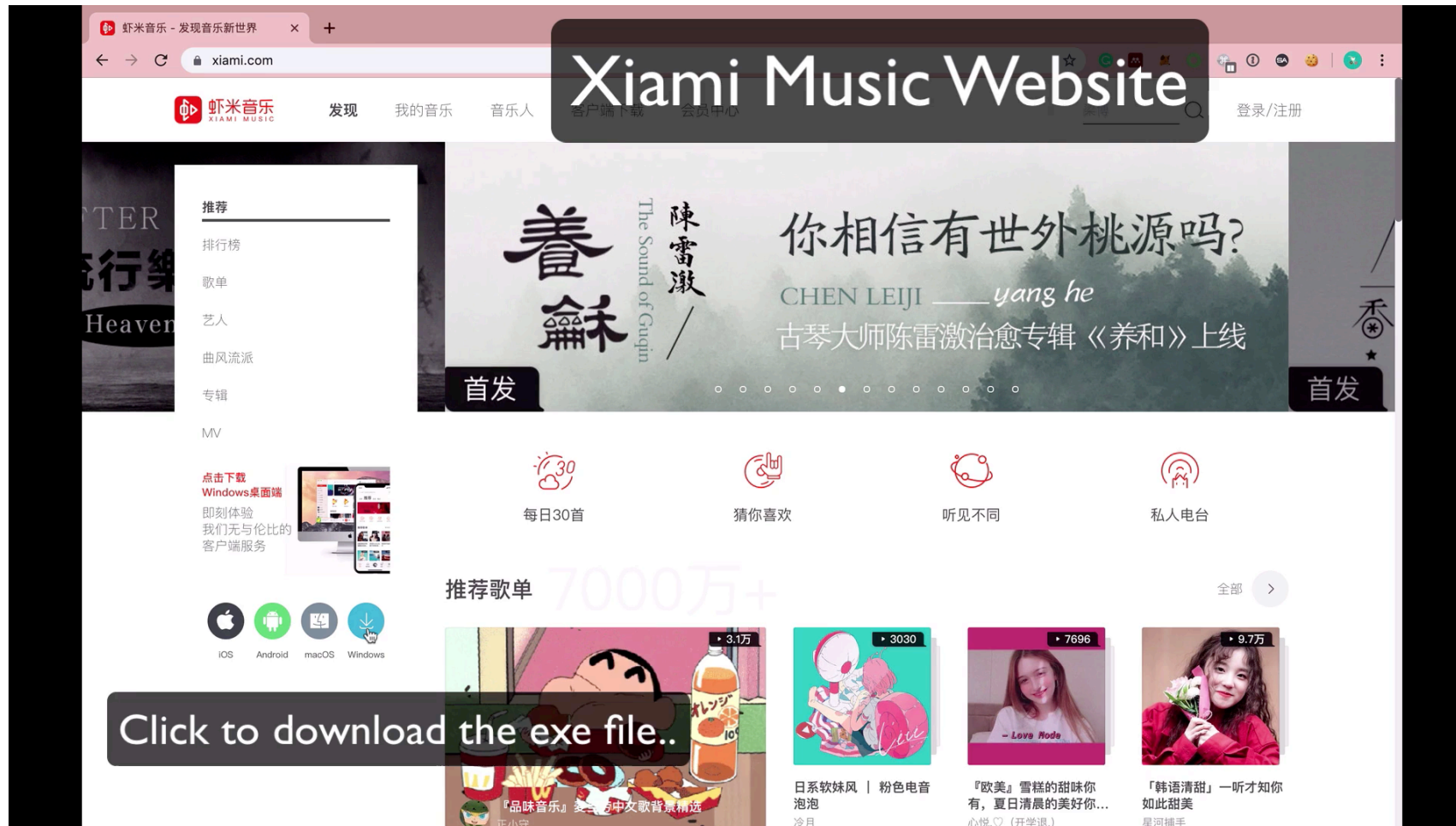
Clear HSTS Policy for ServerA (HSTS-1)

Cancel HSTS Policy for ServerA's Subdomains (HSTS-2)

Decrease HSTS Validity Period for ServerA (HSTS-3).

Real-world Attacks

1. Downgrade a new HTTPS connection.

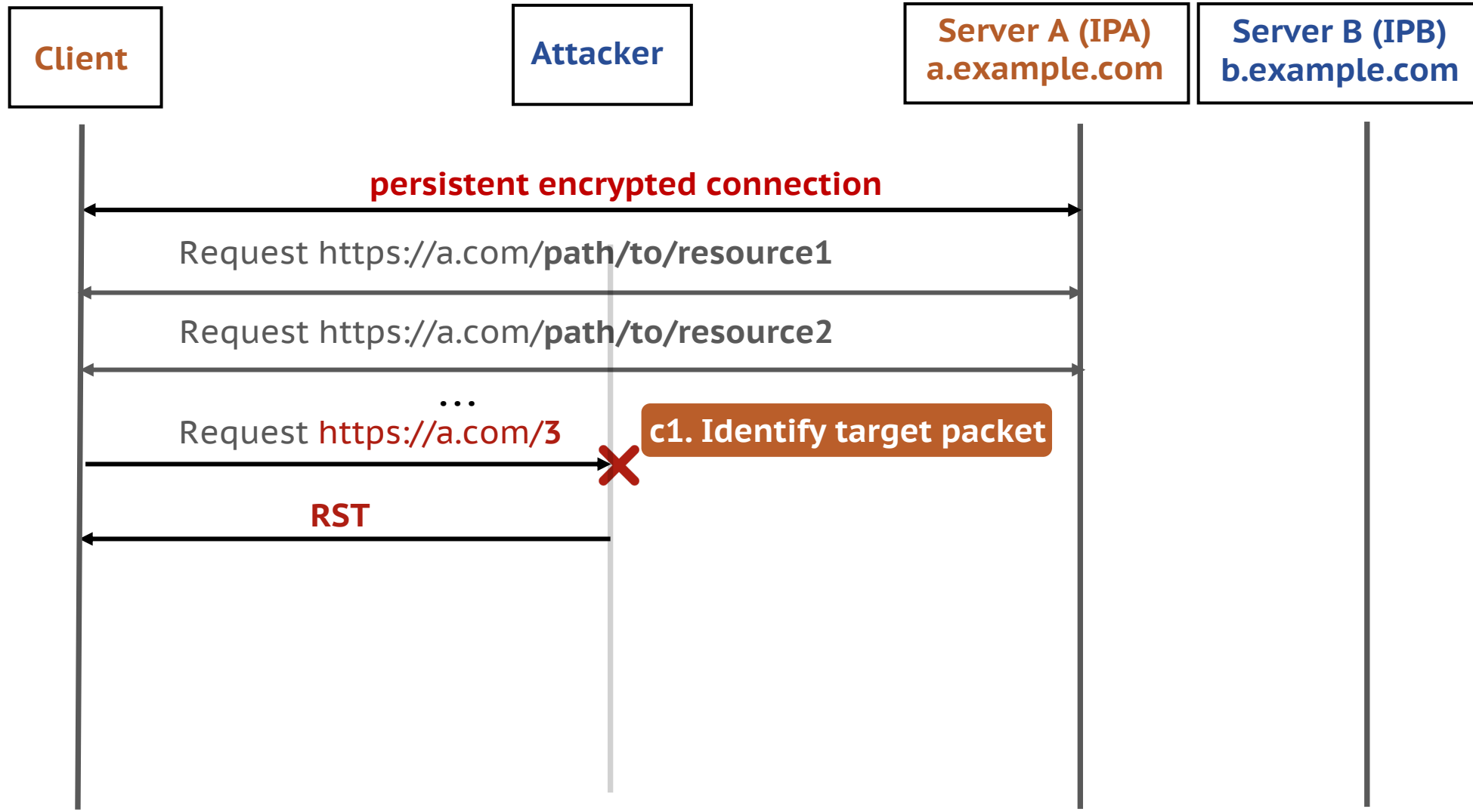


Attacker replaces the download file.

Xiami Music Website is a freemium music streaming services owned by Alibaba Group

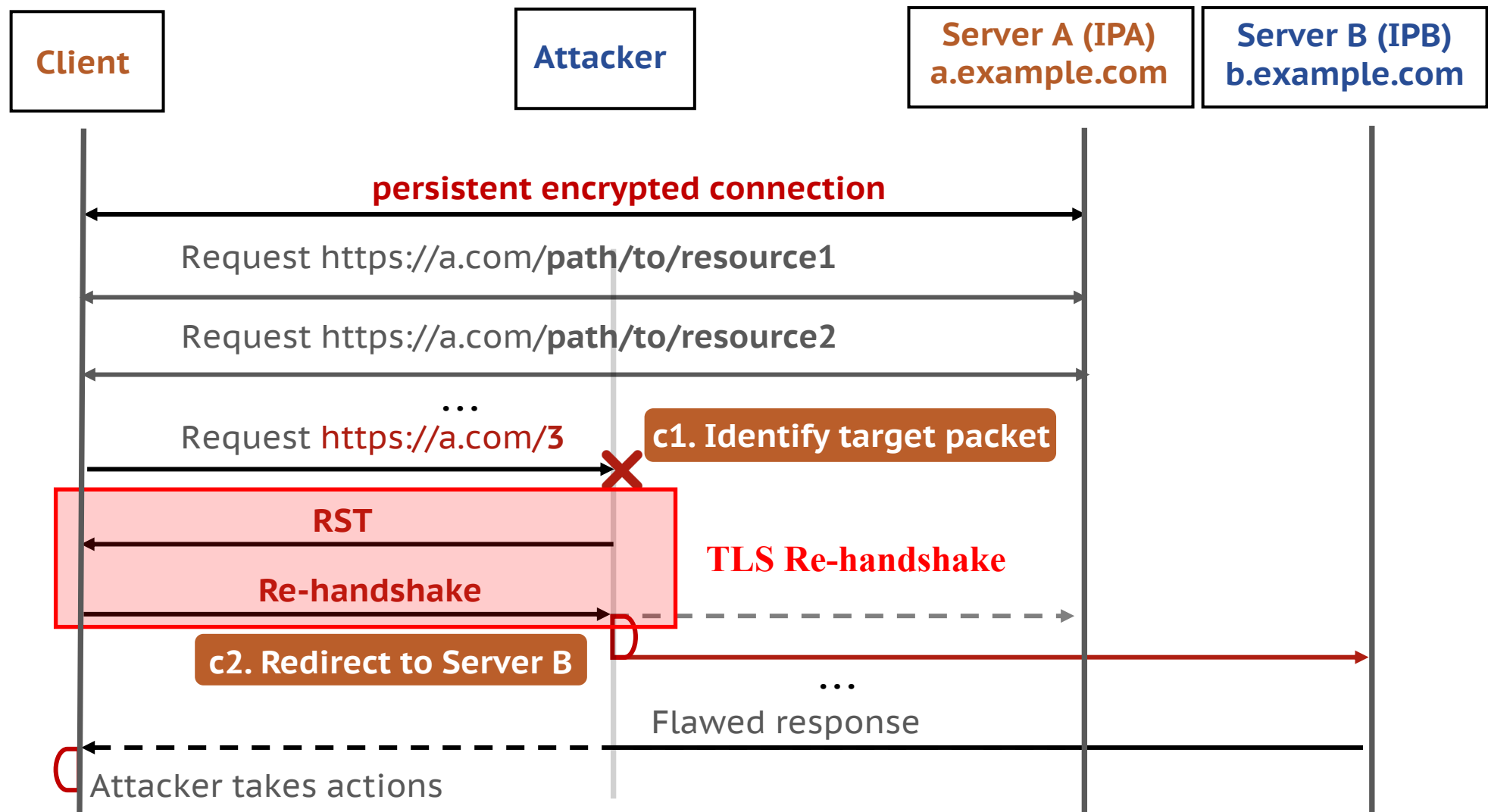
Real-world Attacks

2. Downgrade an already-established HTTPS connection



Real-world Attacks

2. Downgrade an already-established HTTPS connection



Real-world Attacks

2. Downgrade an already-established HTTPS connection

- TLS Re-handshake (triggered by **TCP RST** or **Timeout**)



Table 1. Browser re-handshake behaviors

Trigger Method	Browser	Windows	MacOS	Linux
RST	Chrome	✓	✓	✓
	Firefox	✓	✓	✓
	Edge	✓	-	-
	Safari	-	✓	-
Timeout	Chrome	✓		
	Firefox	✓		
	Edge		-	-
	Safari	-		-

The cases with ✓ can be exploited by attackers to trigger re-handshakes successfully.

Vulnerable Servers in the Wild

- Measurement on **Alexa Top 500 domains** and all their **subdomains**

Finding 1: 2,918 (8.50%) subdomains under 126 (25.2%) Alexa Top 500 base domains are vulnerable to SCC attacks.

Affected Apex Domain Names			
Attack Type		Count	Total
HTTPS Downgrade	Down-1	114 (22.8%)	126 (25.2%)
	Down-2	24 (5.4%)	
HSTS Bypass	HSTS-1	5 (1%)	
	HSTS-2	21 (4.2%)	
	HSTS-3	31 (6.2%)	

Vulnerable Servers in the Wild

- Measurement on **Alexa Top 500 domains** and all their **subdomains**

Finding 2: Popular applications could be affected by SCC attacks.

Possible Attacks

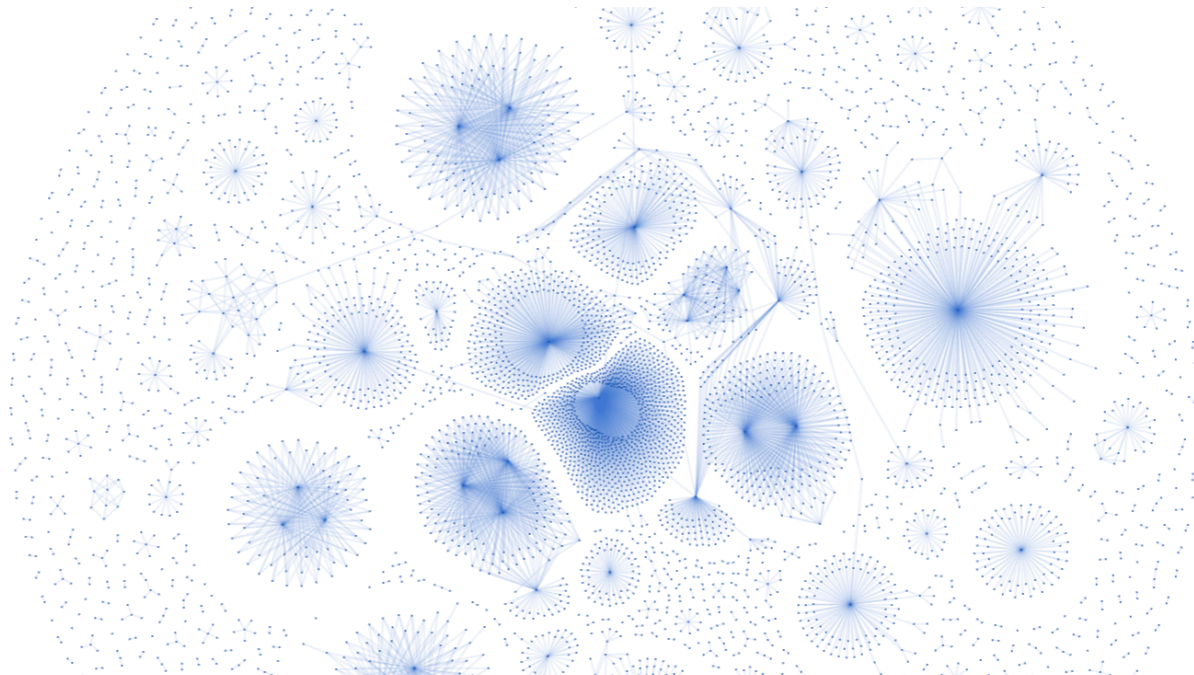
- Online Payment Hijacking
- Download Hijacking
- Website Phishing



Vulnerable Servers in the Wild

- Measurement on **Alexa Top 500 domains** and all their **subdomains**

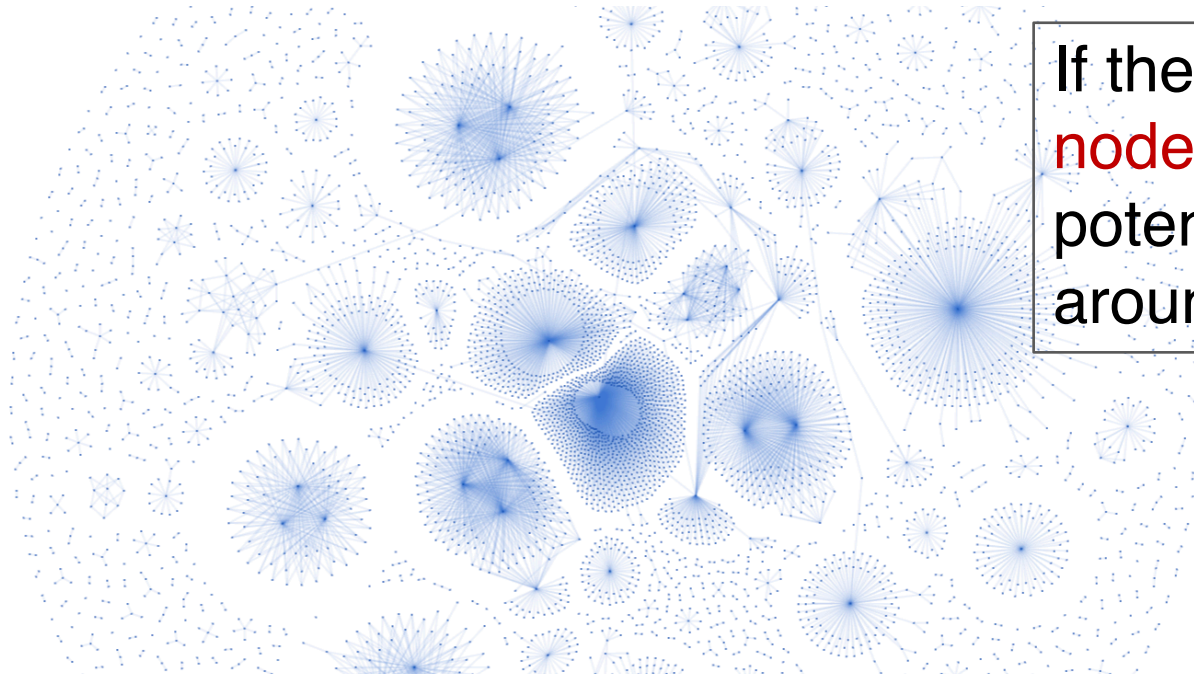
Finding 3: Certificate Sharing is prevalent, which could be vulnerable due to security dependencies among domains.



Vulnerable Servers in the Wild

- Measurement on **Alexa Top 500 domains** and all their **subdomains**

Finding 3: Certificate Sharing is prevalent, which could be vulnerable due to security dependencies among domains.

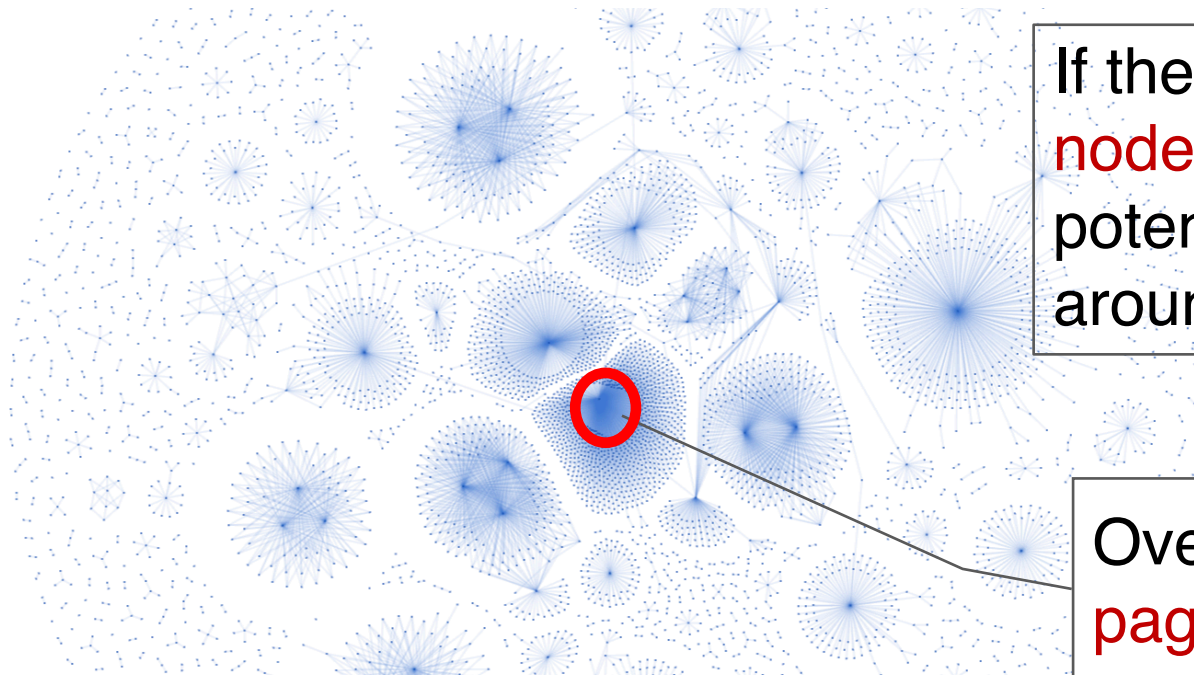


If the domains at the **convergent nodes** are vulnerable, there will be potential security threats for those around them.

Vulnerable Servers in the Wild

- Measurement on **Alexa Top 500 domains** and all their **subdomains**

Finding 3: Certificate Sharing is prevalent, which could be vulnerable due to security dependencies among domains.



If the domains at the **convergent nodes** are vulnerable, there will be potential security threats for those around them.

Over **900** FQDNs depend on **pages.ebay.com**.

Discussion

- **Root Causes**

- **Security dependencies** caused by Certificate Sharing.
- **Problematic implementations** of security policies among different parties.

- **Mitigation**

- Add a notification for the insecure changes of context.
- Well-configure the security policies (e.g., HSTS, CSP, Default 302 Redirect).
- Block all mixed contents. (e.g., plans of Chrome¹ and Firefox²)

¹ <https://www.gsqi.com/marketing-blog/google-chrome-block-mixed-content/>

² <https://support.mozilla.org/en-US/kb/mixed-content-blocking-firefox>

Thank You.

Q & A

{zmm18, zxf19}@mails.tsinghua.edu.cn