Personalized Pseudonyms for Servers in the Cloud

Qiuyu Xiao (UNC-Chapel Hill)
Michael K. Reiter (UNC-Chapel Hill)
Yinqian Zhang (Ohio State Univ.)
Background

Server’s identity is not well protected with the normal HTTPS connection.

DNS query
Query name: www.example.com

IP address: 111.111.111.111
SNI: example.com
Certificate subject name: example.com
Pub key: E5B2C78

Encrypted payload
Background

Server’s identity is not well protected with the normal HTTPS connection.

DNS query
Query name: www.example.com

TCP
TLS/SSL
Encrypted payload

SNI: example.com
Certificate subject name: example.com
Pub key: E58B2C78.....

IP address: 111.111.111.111
Background

Real-world adversaries compromise user’s privacy.
Background

Real-world adversaries compromise user’s privacy.
Background

Real-world adversaries compromise user’s privacy.
Existing solutions

- VPN tunneling
  - Encrypt and tunnel user’s traffic through proxy server
Existing solutions

- Tor
  - Route encrypted packets through multiple Tor relays
Existing solutions

• Cloud and CDN based solutions
  - CloudTransport\(^1\)
  - Domain fronting\(^2\)
  - CacheBrowser\(^3\)

2. Blocking-resistant communication through domain fronting, PETS 2015
3. CacheBrowser: Bypassing Chinese censorship without proxies using cached content, CCS 2015
Existing solutions

- Cloud and CDN based solutions
  - CloudTransport\([1]\)
  - Domain fronting\([2]\)
  - CacheBrowser\([3]\)

2. Blocking-resistant communication through domain fronting, PETS 2015
3. CacheBrowser: Bypassing Chinese censorship without proxies using cached content, CCS 2015
Existing solutions

- Cloud and CDN based solutions
  - CloudTransport\textsuperscript{[1]}
  - Domain fronting\textsuperscript{[2]}
  - CacheBrowser\textsuperscript{[3]} \quad \rightarrow \text{Domain name is visible in TLS SNI field}

2. Blocking-resistant communication through domain fronting, PETS 2015
3. CacheBrowser: Bypassing Chinese censorship without proxies using cached content, CCS 2015
Our solution

**Personalized Pseudonym for a Server in the Cloud (PoPSiCl)**

- **DNS query**
  - Query name: www.example.com

- **IP**
  - TCP
  - TLS/SSL
  - Encrypted payload
    - SNI: example.com
    - Certificate subject name: example.com
    - Pub key: E58B2C78.....
    - IP address: 111.111.111.111
Our solution

**Personalized Pseudonym for a Server in the Cloud (PoPSiCl)**

- **DNS query**
  - Query name: www.example.com

- **IP**
- **TCP**
- **TLS/SSL**
- **Encrypted payload**
  - SNI: example.com
  - Certificate subject name: example.com
  - Pub key: E58B2C78.....
  - IP address: 111.111.111.111
Our solution

**Personalized Pseudonym for a Server in the Cloud (PoPSiCl)**

- **DNS query**: Query name: `x...x.popsicls.com`
- **IP**: `124.132.215.121`
- **SNI**: `x...x.popsicls.com`
- **Certificate subject name**: `x...x.popsicls.com`
- **Pub key**: AGJ46DM.....
- **IP address**: `124.132.215.121`
Our solution

Personalized Pseudonym for a Server in the Cloud (PoPSiCl)

DNS query
Query name: x...x.popsicls.com

IP
TCP
TLS/SSL
Encrypted payload

SNI: x...x.popsicls.com
Certificate subject name: x...x.popsicls.com
Pub key: AGJ46DM.....
IP address: 124.132.215.121

No extra client application!
Our solution

**Personalized Pseudonym for a Server in the Cloud (PoPSiCl)**

- **DNS query**
  - Query name: x...x.popsicls.com

- **IP**
  - TCP
    - TLS/SSL
      - Encrypted payload
        - SNI: x...x.popsicls.com
        - Certificate subject name: x...x.popsicls.com
        - Pub key: AGJ46DM.....

- **IP address:** 124.132.215.121

- No proxy!

- No extra client application!
Threat model

In the context of a client-server interaction ...

• What is trusted
  • Client computer
  • Cloud infrastructure (including the server computer)

• What is not trusted
  • The network between the client and the cloud
  • Other clients and other servers
PoPSiCl registration

Welcome to PoPSiClis!

I want to visit www.example.com privately.

Go!
PoPSiCl registration

Welcome to PoPSiCls!

https://www.popsicles.com/goto?domain_name=www.example.com

Registration done!

Your PoPSiCl for **www.example.com** is **ff5avqyu.hw9wt.popsicles.com**?capability=bdsdFaXkmLg@ze.

Please download the client cert and private key [here](#).
PoPSiCl registration

PoPSiCl store

DNS server

SDN controller

Cloud
PoPSiCl registration

Registration request → PoPSiCl store

SDN controller

DNS server

Cloud
PoPSiCl registration

PoPSiCl store

DNS server

SDN controller

Cloud

PoPSiCl

Pseudo IP
PoPSiCl registration

PoPSiCl

Tenant server ID

PoPSiCl store

DNS server

SDN controller

PoPSiCl

Pseudo IP

VM

VM

VM

VM

VM
PoPSiCl registration

- Client Cert
- Server Cert
- Client PriKey
- Server PriKey

PoPSiCl store

PoPSiCl

Pseudo IP

DNS server

PoPSiCl

Tenant server ID

SDN controller

VM

Cloud
PoPSiCl registration

- Sign
- Server PriKey
- Client Cert
- Client PriKey

PoPSiCl store

- Client Cert
- Server Cert

PoPSiCl

- Pseudo IP

DNS server

PoPSiCl

Tenant server ID

SDN controller

Cloud

VM

VM

VM

VM
PoPSiCl registration

- PoPSiCl
- Client Cert
- Server Cert
- Client PriKey
- Server PriKey

Client

PoPSiCl store

PoPSiCl

Pseudo IP

DNS server

Tenant server ID

SDN controller

VM

VM

VM

VM

Cloud
PoPSiCl registration

PoPSiCl store

SDN controller

DNS server

Tenant server ID

Pseudo IP

Server Cert

Server PriKey

Client Cert

Client PriKey
PoPSiCl registration

PoPSiCl
Client Cert
Client PriKey

PoPSiCl store

DNS server
Tenant server ID
SDN controller

PoPSiCl
Pseudo IP

Server Cert
Server PriKey

VM
VM
VM
VM
PoPSiCl access

- DNS server
- SDN switch
- SDN controller
- PoPSiCl Server Cert
- Server PriKey
- VM
- Tenant server ID
- PoPSiCl Pseudo IP
- Client Cert
- Client PriKey
PoPSiCl access

(1) DNS query: PoPSiCl

PoPSiCl Client Cert
Client PriKey

PoPSiCl Server Cert
Server PriKey

PoPSiCl Pseudo IP

DNS server

SDN switch

SDN controller

VM

Tenant server ID

Cloud
PoPSiCl access

(1) DNS query: PoPSiCl

(2) DNS response: Pseudo IP

PoPSiCl Client Cert
PoPSiCl Client PriKey

PoPSiCl Pseudo IP

DNS server

SDN switch

PoPSiCl Server Cert
PoPSiCl Server PriKey

SDN controller

PoPSiCl Tenant server ID

Cloud
PoPSiCl access

(1) DNS query:
PoPSiCl
Client Cert
Client PriKey
DNS server
PoPSiCl
Pseudo IP
(2) DNS response:
Pseudo IP
(3) Pseudo IP
SDN switch
SDN controller
PoPSiCl
Server Cert
Server PriKey
Cloud
PoPSiCl access

(1) DNS query: PoPSiCl

(2) DNS response: Pseudo IP

(3) Pseudo IP

(4) Forward

PoPSiCl Client Cert
PoPSiCl Client PriKey

PoPSiCl Server Cert
PoPSiCl Server PriKey

SDN controller

SDN switch

Cloud

VM VM VM VM

PoPSiCl Pseudo IP

Tenant server ID

VM

Cloud

11
PoPSiCl access

1. DNS query: 
   - PoPSiCl
   - Client Cert
   - Client PriKey
   - Tenant server ID

2. DNS response: 
   - PoPSiCl
   - Pseudo IP
   - Server Cert
   - Server PriKey

3. Pseudo IP

4. Forward

5. Establish TCP (via SDN switch)
PoPSiCl access

(1) DNS query: PoPSiCl

(2) DNS response: Pseudo IP

Get PoPSiCl from the SNI field in TLS ClientHello message.

(3) Pseudo IP

(4) Forward

(5) Establish TCP (via SDN switch)

DNS server

PoPSiCl

Client Cert

Client PriKey

PoPSiCl

PoPSiCl

Pseudo IP

SDN switch

PoPSiCl

Server Cert

Server PriKey

PoPSiCl

Tenant server ID

Cloud

11
(1) DNS query: PoPSiCl

(2) DNS response: PoPSiCl

(3) Pseudo IP

(4) Forward

(5) Establish TCP (via SDN switch)

(6) Rule update
PoPSiCl access

<table>
<thead>
<tr>
<th>MATCH</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>Source port</td>
</tr>
<tr>
<td>Client-IP</td>
<td>Client-port</td>
</tr>
<tr>
<td>Tenant-IP</td>
<td>Server-port</td>
</tr>
</tbody>
</table>

(1) DNS query:

(2) DNS response:

(3) Pseudo IP

(4) Forward

(5) Establish TCP (via SDN switch)

(6) Rule update

PoPSiCl
Client Cert
Client PriKey

PoPSiCl Pseudo IP

Server Cert
Server PriKey

SDN switch

SDN controller

Tenant server ID

Cloud
PoPSiCl access

1. DNS query: PoPSiCl
2. DNS response: PoPSiCl
3. Pseudo IP
4. Forward
5. Establish TCP (via SDN switch)
6. Rule update
7. TCP hand-off
PoPSiCl access

(1) DNS query: PoPSiCl

(2) DNS response: PoPSiCl

(3) Pseudo IP

(4) Forward

(5) Establish TCP (via SDN switch)

(6)(8) Rule update

(7) TCP hand-off

PoPSiCl Client Cert
PoPSiCl Client PriKey

PoPSiCl Server Cert
PoPSiCl Server PriKey

VM

SDN switch

PoPSiCl Tenant server ID

Cloud
PoPSiCl access

<table>
<thead>
<tr>
<th>MATCH</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>Source port</td>
</tr>
<tr>
<td>Client-IP</td>
<td>Client-port</td>
</tr>
<tr>
<td>Tenant-IP</td>
<td>Server-port</td>
</tr>
</tbody>
</table>

1. DNS query:
2. DNS response:
3. Pseudo IP
4. Forward
5. Establish TCP (via SDN switch)
6. Rule update
7. TCP hand-off
8. Rule update
PoPSiCl access

<table>
<thead>
<tr>
<th>MATCH</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>Source port</td>
</tr>
<tr>
<td>Client-IP</td>
<td>Client-port</td>
</tr>
<tr>
<td>Tenant-IP</td>
<td>Server-port</td>
</tr>
</tbody>
</table>

1. DNS query:
2. DNS response:
3. Pseudo IP
4. Forward
5. Establish TCP (via SDN switch)
6. Rule update
7. TCP hand-off
8. Rule update

PoPSiCl Client Cert
Client PriKey

SDN switch

Tenant server ID

PoPSiCl

Server Cert
Server PriKey

Cloud
PoPSiCl access

(1) DNS query: PoPSiCl
(2) DNS response: Pseudo IP
(3) Pseudo IP
(4) Forward
(5) Establish TCP (via SDN switch)
(6)(8) Rule update
(7) TCP hand-off
(9) TLS (via SDN switch)

PoPSiCl Client Cert
Client PriKey
PoPSiCl Server Cert
Server PriKey

DNS server
SDN switch
SDN controller

VM
VM
VM
VM

PoPSiCl Tenant server ID

Cloud
PoPSiCl access

Accept the connection only if the user can present a valid Client Cert

(1) DNS query: PoPSiCl
(2) DNS response: Pseudo IP
(3) Pseudo IP
(4) Forward
(5) Establish TCP (via SDN switch)
(6)(8) Rule update
(7) TCP hand-off
(9) TLS (via SDN switch)

PoPSiCl Client Cert
Client PriKey

PoPSiCl Server Cert
Server PriKey

SDN controller

PoPSiCl Tenant server ID

Cloud
Implementation

Cloud
- OpenStack-based IaaS cloud deployed in CloudLab testbed
- PoPSiCl store and SDN controller are implemented in C and C++
- Open vSwitch as the SDN switch in each physical machine

Tenant server
- A Linux kernel module for TCP state transfer
- Each PoPSiCl is mapped to a virtual host in Nginx server
Latency
Latency

![Graph showing download latency vs. web object size for HTTPS connection.](image_url)
Latency

![Latency Graph]
Latency

![Graph showing latency comparison between HTTPS, PoPSiCl, and Tor based on web object size (1KiB, 10KiB, 100KiB, 1MiB, 5MiB). The x-axis represents the web object size, and the y-axis represents download latency (in ms). The graphs show that Tor and PoPSiCl have lower latency than HTTPS, especially for larger web object sizes.](image-url)
Latency
Throughput

![Throughput Graph](image-url)
Throughput

![Graph showing the relationship between request rate (req./s) and response rate (resp./s) for HTTPS.]
Throughput
Throughput

![Graph showing throughput as a function of request rate. The graph compares HTTPS, HTTPS with client auth., and PoPSiCl.](image)
Scalability: Throughput per retrieved object size
Scalability: Throughput per retrieved object size

[Graph showing the relationship between web object size and maximum throughput per response/s, with a downward trend indicating decreasing throughput as object size increases.]
Scalability: Throughput per retrieved object size

![Diagram showing scalability with throughput per retrieved object size against web object size (Kib)].
Scalability: Latency per # switch rules
Scalability: Latency per # switch rules
Scalability: Latency per # switch rules
Scalability: Latency per # PoPSiCls for one server
Scalability: Latency per # PoPSiCls for one server
Scalability: Latency per # PoPSiCls for one server

![Graph showing scalability with latency per PoPSiCls for one server]

Latency (ms)

Number of virtual hosts
Q&A