# CDN Judo : Breaking the CDN DoS Protection with Itself

Accepted by the ISOC Network and Distributed System Security Symposium (NDSS), San Diego, CA, Feb. 2020.

Kaiwen Shen





#### **Network acceleration**

- Infrastructure for access acceleration and DoS defense
  - ➤ 38.98% of top 10K websites use CDN [Your Remnant Tells Secret-DSN'18]
  - ➤ We find CDN itself can be abuse to break its DoS protection



#### **Dos attack**

- Infrastructure for access acceleration and DoS defense
  - ➤ 38.98% of top 10K websites use CDN [Your Remnant Tells Secret-DSN'18]
  - ➤ We find CDN itself can be abuse to break its DoS protection



#### **Content Delivery Network**

- Infrastructure for access acceleration and DoS defense
  - ➤ 38.98% of top 10K websites use CDN [Your Remnant Tells Secret-DSN'18]
  - ➤ We find CDN itself can be abuse to break its DoS protection



#### **CDN Forwarding Process**



## **Our Work**

#### Exploiting CDN forwarding features to attack the origin

Attack-1	HTTP/2 amplification attack
Attack-2	Pre-POST slow HTTP attack
Attack-3	Egress IP blocking attack

#### Performed real-world evaluations on six vendors



#### Attack-1

## **HTTP/2 Amplification Attack**

## **HTTP/2 Protocol**

- Designed to improve HTTP performance
   » RFC7540, released in 2015
- \* Compression (to reduce header redundancy)
  - \* Binary protocol, HPACK header compression
- Connection reuse (to reduce TCP connections)
  - Request -> Stream
  - \* Streams are multiplexed

Deployment: Over 43.2% of Alexa top 1M websites (w3techs.com, 12 Feb 2020)

## **Concept of HTTP/2 Amplification attack**

#### ✤ Our study

- >Identify that HTTP/2-1.1 conversion of CDN will cause amplification attack.
- >Improve the attack with the feature of Huffman encoding.
- >Real-world measurement and evaluation



[HTTP/2 Tsunami Attack, EST '17]
 Show bandwidth amplification attack in local proxies built with Nginx and Nghttp2.

#### **HPACK Static Table**

An indexed table of common header fields
pre-defined in both HTTP/2 client and server.



## Attack-1.1: Using HPACK Static Table

✤ HTTP/2-1.1 conversion of CDN causes a bandwidth amplification.



#### Bandwidth amplification factor: 49B / 11B = 4.45

## HPACK Dynamic Table (1/2)

 An indexed table of previously seen headers to avoid repeatedly transferring headers.

>Step 1: The firstly seen headers will be inserted into the dynamic table.



## HPACK Dynamic Table (2/2)

 An indexed table of previously seen headers to avoid repeatedly transferring headers.

>Step 2: The subsequently repeated headers will be substituted as an index.



## Attack-1.2: Using HPACK Dynamic Table

\* The dynamic table enhances this kind of bandwidth amplification.



Bandwidth amplification factor:  $4039B \times (N+1) / 3999B + 5B \times N = \frac{4039 + 4039N}{3999 + 5N}$ For example, when N is 100, the factor is 88.70.

## **Attack-1.3: Improve with Huffman Encoding**

- Some special characters can have short Huffman encodings
  - >The Huffman encoding of 'X' is 8 bits in length.
  - ≻Characters {0, 1, 2, a, c, e, i, o, s, t} have the shortest Huffman encoding (5 bits).



## **Attack-1.3: Improve with Huffman Encoding**

✤ The shorter the Huffman encoding, the larger the amplification factor.

	Huffman Encoding Length	<b>Amplification Factor</b>		
Character 'X'	8 bits	$\frac{4039 + 4039 \text{N}}{3999 + 5 \text{N}}$	88.70 when N is 100	
Character 'a'	5 bits	$\frac{4039 + 4039 \text{N}}{2511 + 5 \text{N}}$	131.13 when N is 100	

Note: N is the concurrent streams in the same HTTP/2 connection.

#### **Bandwidth Amplification Evaluation**

- Create multiple concurrent requests in one HTTP/2 connection.
  - > The amplification factor grows with the number of concurrent streams.
  - >The max factor is got at the position of the max concurrent streams.



#### **Comparison with previous work**

✤ Our work achieved larger amplification factors than previous work.

	Max Streams	100		128			256
	Evaluation Platform	MaxCDN	Fastly	CDNsun	CloudFront	KeyCDN	Cloudflare
Our Attack	Amplification Factor	94.7	97.9	118.7	116.9	105.5	166.1
HTTP/2 Tsunami Attack	Evaluation Platform	HTTP/2 Proxies built with Nginx and Nghttp2					
THURK	Amplification Factor	79.2		94.4			140.6

#### **HTTP/2 Connection Amplification Attack**

♦ concurrent streams in one HTTP/2 connection  $\rightarrow$  multiple HTTP/1.1 connections



	CloudFront	Cloudflare	CDNSun	Fastly	KeyCDN	MaxCDN
Max concurrent streams per HTTP/2 connection	128	256	128	100	128	100
<b>Connection</b> <b>Amplification</b>	Yes	Yes	-	-	-	Yes



#### Mitigation

Threats	Recommendation
HTTP/2 attack	HTTP/2 support for back-end connection limit the back-end network traffic.
Pre-POST attack	limit the number of CDN back-to-origin connections enforce strict forwarding (store-then-forward).
Egress IP blocking	apply unpredictable egress IP churning strategy.

#### **Responsible Disclosure**

- **Cloudflare:** reproduced HTTP/2 amplification with 126x and rewarded us \$200 bonus.
- \* Fastly: Confirmed our report and offered us T-shirts.
- CloudFront: suggested HTTP/2 amplification is a feature of HTTP/2 standard, and would like to use rate-based WAF rules to mitigate the attack.
- MaxCDN: stated the egress IP blocking is out of scope as it involves with additional GFW infrastructure.
- \* **CDNSun** and **KeyCDN**: received our report and but no further comments so far.

## Summary

- \* A empirical security study on CDN back-end connections
  - HTTP/2 amplification attack
  - \* pre-POST slow HTTP attack
  - Egress IP blocking attack
- Real-world evaluation on six CDN vendors
  - Received positive feedback from some CDNs
- How to balance performance and security

Thank you!