China Telecom Cybersecurity Tech | 中国联通 China unicom | 联通数科 Unicom Digital Tech | Baidu Security | OASES | NEXUSGUARD DDoS Protection Made Simple | HUAWEI

Global DDoS Attack Status and Trend Analysis in 2022

Jointly released by China Telecom Cybersecurity Technology Co.,Ltd., China Unicom Digital Technology Company Limited, Baidu Security, Nexusguard, and Huawei.

Scan to download

Contents

01 Summary

1.1 Expert Opinions

Opinion 1: The intensity and frequency of DDoS attacks targeting the finance industry keep increasing, challenging the performance of WAFs. To address this, financial enterprises need to build a three-layer defense architecture — consisting of carriers' upstream cloud mitigation services, anti-DDoS systems deployed at the borders of the enterprise private clouds, and WAFs — to mitigate application-layer attacks.

Opinion 2: APIs have become a new target of DDoS attacks. Therefore, a DDoS protection architecture for APIs needs to be improved.

1.2 A Famous DDoS Attack Incident

During the 2022 World Cup, a payment platform in China suffered large-scale high-intensity DDoS attacks, which lasted for five days. During the attacks, the attack intensity and sophistication kept increasing, and reached almost the highest in the DDoS attack-defense confrontation history. The peak attack rate exceeded 800 Gbps for seven times, 1 Tbps for four times, and peaked at 1.159 Tbps. The entire attack process can be divided into seven phases. Attackers continuously launched high-strength network-layer CC to challenge the defense success rate and used network-layer attacks, such as UDP flood, UDP fragment, UDP reflection, TCP reflection, large SYN, and other flood attacks to overload the link bandwidth and increase defense costs. In the last round of attack, SYN attacks with real source IP addresses and application-layer CC joined for the last battle. However, even with various types of attack tactics used, hackers failed to break through the defense line.

1.3 DDoS Attack Situation

Ultra-large scale attacks are extremely active.

In 2022, terabit-strong attacks were extremely active. There were 232 attacks above 800 Gbps, 1.67 times that in 2021. In April, China Telecom's security team detected the attack with the annual highest packet rate, which peaked at 861 Mpps. In November, China Telecom's security team detected the attack with the annual maximum bandwidth, which peaked at 3.189 Tbps. In June 2022, the largest application-layer attack in the history of the Internet occurred, with a peak attack rate of 46 million requests per second (rps)[1]. The class C IP address segments hit by carpet-bombing attacks increased from 100+ in 2021 to 600+.

Summary

The attack frequency continues to increase.
The attack frequency in 2022 was 1.1 times that in 2021 and 1.4 times that in 2020.

Traffic of volumetric attacks surges in seconds, and the ramp-up speed reaches a record high, challenging the response speed of the defense system.
Terabit-strong attacks feature fast flooding [1], which is a noticeable signature of volumetric attacks. In 2021, it took 20 seconds for peak attack traffic to ramp up to a range of 800 Gbps to 1 Tbps, but in 2022, this took only 10 seconds.

Evolved attack complexity and new variants of network-layer CC are challenging the defense success rate. HTTP/2 multiplexing is abused to launch application-layer attacks peaking at tens of millions of requests per second, increasing the defense cost. Escalated carpet-bombing attacks use content security devices to launch HTTP reflection and RST carpet-bombing attacks. To address the challenges, security products need to improve self-defense abilities.
CC attacks at the network layer and spoofed QUIC flood attacks (a type of UDP flood targeting port 443) are active. In the past three years, the proportions of ACK flood and UDP flood attacks in network-layer attacks have been increasing year by year. In 2022, ACK flood attacks accounted for 23.16%, 2.1 times that in 2021 and 10.9 times that in 2020. In 2022, UDP flood attacks accounted for 21.26%, 1.3 times that in 2021 and 1.6 times that in 2020.

The network-layer CC attacks targeting HTTP and HTTPS service ports are also active. As a result, in 2022, the frequency of HTTP abnormal session attacks (20.81%) and TLS abnormal session attacks (23.98%) increased compared with that in 2021.

New variants of network-layer CC attacks emerge. Network-layer CC attacks launched by the Mirai botnet combine techniques related to session attacks and those related to flood attacks with forged source IP addresses. After a TCP connection is established between a zombie and a server through a socket, ACK packets are sent through a raw socket on the same connection, consuming server performance and overloading the inbound bandwidth. To improve the evasion capability, ACK flood attacks that carry complete HTTPS sessions appeared. This type of attack replays ACK packets quickly to trigger retransmissions on a server, occupying outbound bandwidth. Then, a new attack tactic emerged: TCP Keep-Alive packets are used to maintain TCP sessions for a long time, exhausting session resources on the server.

HTTP/2 introduces the multiplexing transmission mechanism based on binary streams, allowing all requests and responses under the same domain name to be efficiently transmitted based on a single TCP connection. In August 2022, Baidu security team detected multiple application-layer CC attacks peaking at tens of millions of requests per second using the HTTP/2 multiplexing feature.

The public cloud is not only the victim of carpet-bombing attacks, but also passively become the source of attacks. An attacker can initiate SYN carpet-bombing attacks with illegal information carried in packet payloads. As a result, content security devices at the public cloud border passively launch outbound HTTP amplification attacks and inbound RST carpet-bombing attacks.

Both network-layer and application-layer attacks use the fast flooding strategy, challenging the automation rate of the defense system.
57.40% of network-layer attacks and 40.49% of application-layer attacks last for ≤ 5 minutes.

The media and Internet, government and public utilities, education, finance, and healthcare industries are the top 5 targeted industries.
According to Unicom Digital Tech's continuous tracking of attack situations, in 2022, the top 5 targeted industries were media and Internet, government and public utilities, education, finance, and healthcare. Their

proportions are 51.57%, 18.90%, 9.62%, 3.87%, and 2.83%, respectively. Compared with 2021, in 2022, the proportion of attacks targeting the education industry, healthcare industry, as well as governments and public utilities increased by 56.6 times, 8.6 times, and 3.6 times, respectively. The Industrial Internet, as an emerging field, becomes the new target of attacks. In 2022, the frequency at which the Industrial Internet was attacked was 18 times that of 2021.

The global attack targets are Asia-Pacific (APAC), Americas (AMER), Latin America (LATAM), as well as Europe, the Middle East and Africa (EMEA), listed by the number of attacks. Zhejiang, Shandong, and Jiangsu are top 3 targeted Chinese provinces.
The top attack targets are APAC, AMER, LATAM, and EMEA by region, which account for 45.51%, 23.42%, 16.44%, and 14.63%, respectively. The top 3 attack targets in China are Zhejiang, Shandong, and Jiangsu provinces, which account for 16.92%, 12.41%, and 11.30% respectively.

1.4 DDoS Botnet Situation

Botnet family distribution: IoT-based and Linux-based botnet families are the most active. The top 3 families ranked by the number of C2 servers are Mirai, Dofloo, and Gafgyt, accounting for 51.75%, 15.36%, and 13.90% respectively.

Number of botnet C2 servers, by region: AMER, APAC, EMEA, and LATAM, accounting for 33.80%, 33.76%, 28.79%, and 3.65% respectively. The top 3 regions with the largest number of C2 servers in China are Hong Kong, Taiwan, and Guangdong, accounting for 32.58%, 10.29%, and 9.73% respectively.

1.5 DDoS Attack Source Situation

Number of zombies, by region: APAC, AMER, EMEA, and LATAM, accounting for 65.46%, 15.63%, 12.36%, and 6.55% respectively. The top 3 regions with the largest number of zombies in China are Henan, Hong Kong, and Guangdong, accounting for 21.22%, 11.34%, and 9.00% respectively.

Number of zombies, by carrier: In China, China Telecom, China Unicom, and China Mobile, in a descending order, have the largest number of zombies on their networks. Outside China, the victims are Amazon, LLC, and Google.

1.6 Typical Attack Analysis

APIs have become a new target of DDoS attacks, and the attack intensity is challenging. Attack tactics are complex and diversified, and new attacks with variable tactics evolve.
During the 2022 World Cup, a payment platform in China suffered large-scale DDoS attacks peaked at 723 Gbps. The attacks mainly consisted of the SYN flood, network-layer CC, and spoofed QUIC flood attacks. The attacks lasted for 11 hours and 50 minutes.

Summary

In addition to traditional network-layer attacks, APIs are also vulnerable to HTTP & HTTPS application-layer attacks and TLS attacks. In 2022, the top 5 types of attacks targeting APIs were network-layer CC, SYN flood, UDP flood, HTTP flood, and UDP reflection attacks. To evade defense, attackers select multiple APIs as attack targets. Moreover, attackers use multiple methods to initiate access requests to the root directory of the target server, challenging the availability of the API service systems with these invalid requests.

The frequency and intensity of attacks targeting Chinese financial enterprises increase continuously. The attacks become more sophisticated with diverse targets (including network infrastructures and applications), decreasing the defense success rate.

According to China Unicom Digital Tech's analysis on DDoS attack situation of Chinese financial enterprises in the past three years, the number of attacked financial enterprises, attack frequency, and attack intensity increase significantly. In 2022, a total of 102 bank, security, and insurance enterprises were attacked by DDoS attacks, 4.3 times that in 2021 and 9.3 times that in 2020. In 2022, 4012 attacks occurred, 3.9 times that in 2021 and 12.9 times that in 2020. In 2022, the strongest attack peaked at 196 Gbps, 3.4 times that in 2021 and 13.1 times that in 2020. Attacks often occur during the service settlement period. The attack targets are portal websites, financial services systems, and DNS servers.

Most attacks last for ≤ 5 minutes, accounting for 82.17% of the total number of attacks in 2022.

From a global perspective, attacks on financial enterprises are becoming increasingly sophisticated, and multi-vector attacks become the mainstream. Security of network infrastructure and applications is threatened. TLS attacks and HTTPS attacks become the norm, and the rate of attacks from a single source IP address are becoming lower. In addition, new types of network-layer CC attacks emerge.

02 A Famous DDoS Attack Incident

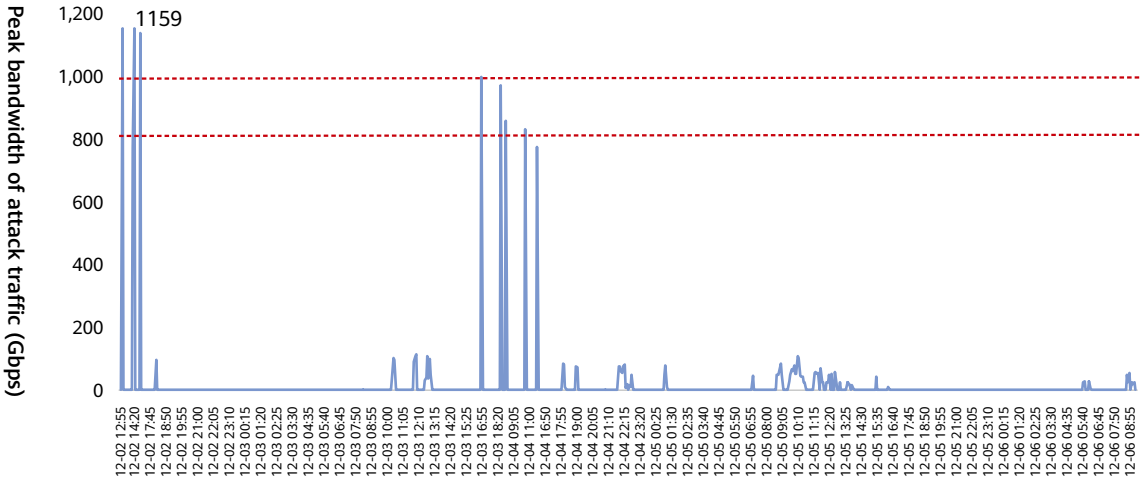
2.1 Overview

During the 2022 World Cup, DDoS attacks were extremely active. Multiple attacks on payment platforms and APIs occurred in China.

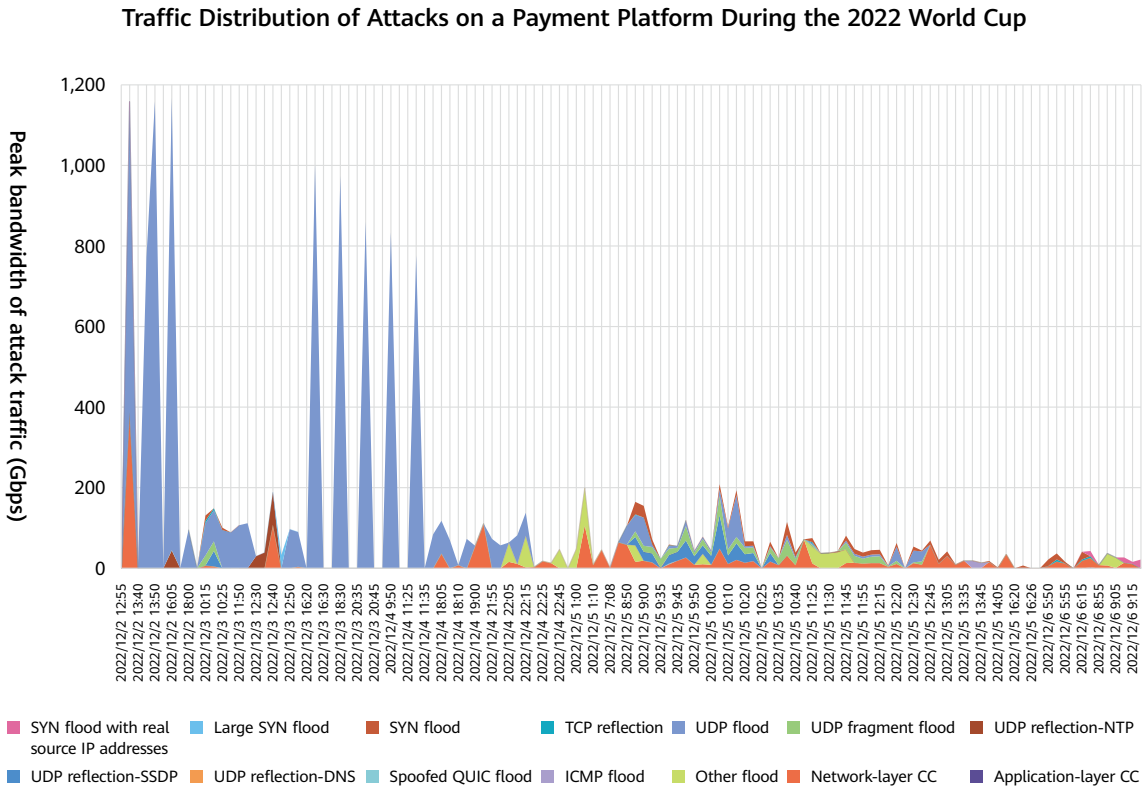
A typical attack is an attack targeting a payment platform, which started at 12:55 on December 2 and peaked at 1.159 Tbps/101.5 Mpps. The entire attack process lasted for five days. Attackers continuously increased the peak bandwidth of attack traffic, and changed attack tactics. Both the attack intensity and attack complexity were almost the highest in the DDoS attack history.

The attack rate exceeded 800 Gbps for seven times and 1 Tbps for four times.

Peak Traffic of Attacks on a Payment Platform During the 2022 World Cup



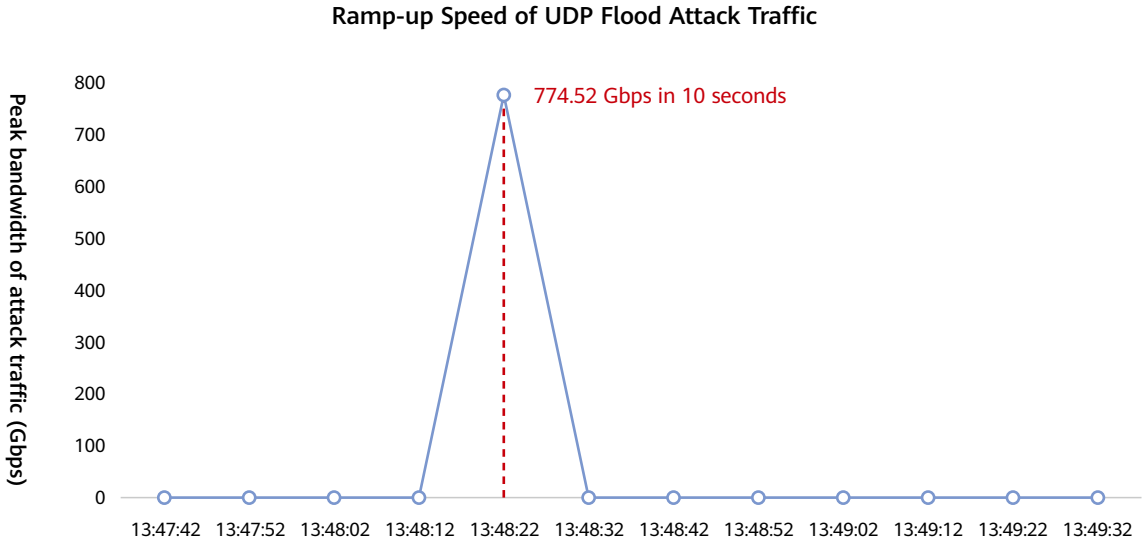
The entire attack process can be divided into seven phases, and a total of 14 attack vectors were used: UDP flood, spoofed QUIC flood, SYN flood, large SYN flood, SYN flood with real source IP addresses, network-layer CC, NTP reflection, SSDP reflection, DNS reflection, UDP fragment, TCP reflection, ICMP flood, other flood, and application-layer CC (HTTP flood).



2.2 Attack Incident Review

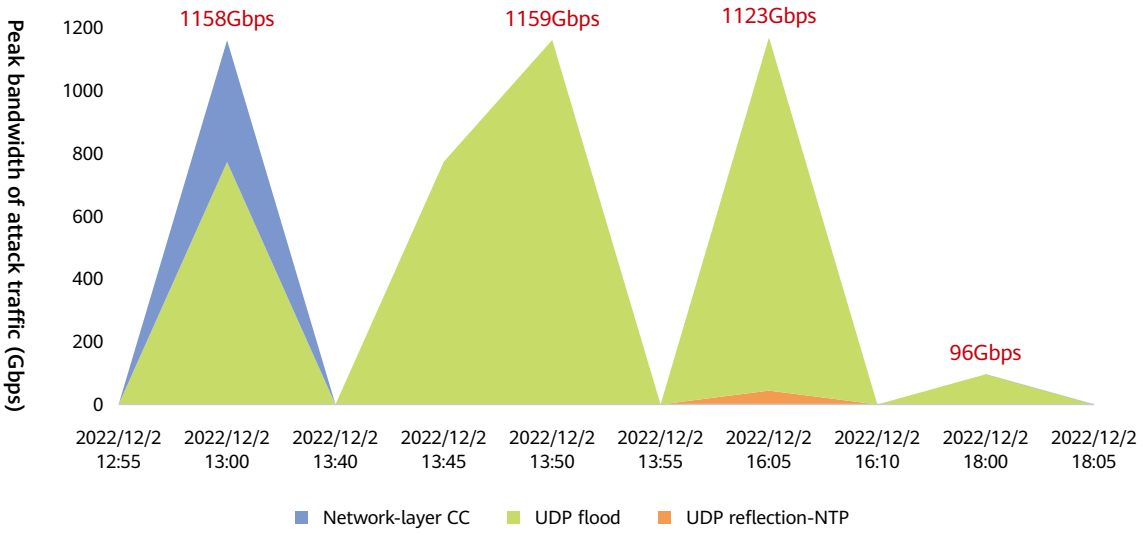
First round of attacks: fast flooding attacks overwhelm the network and challenge the defense response speed.

The first round of attacks started at 12:55 on December 2 and lasted for 5 hours and 5 minutes. The attacker attempted to consume the network bandwidth with high-instensity fast flooding. In this round, UDP flood attacks surged in seconds — reached 774.52 Gbps within 10 seconds, and peaked at 1.123 Tbps.



The first round of attacks was dominated by UDP flood attacks and supplemented by network-layer CC and NTP reflection attacks. In network-layer CC attacks, ACK flood attacks with packets of more than 1000 bytes took the lead. The attack traffic peaked at 387 Gbps, and the attack traffic was mainly from regions outside China. After this round of attacks, the attacked ISPs requested the carrier to enable the policy of blocking traffic outside China, effectively mitigating network-layer CC attacks launched outside China.

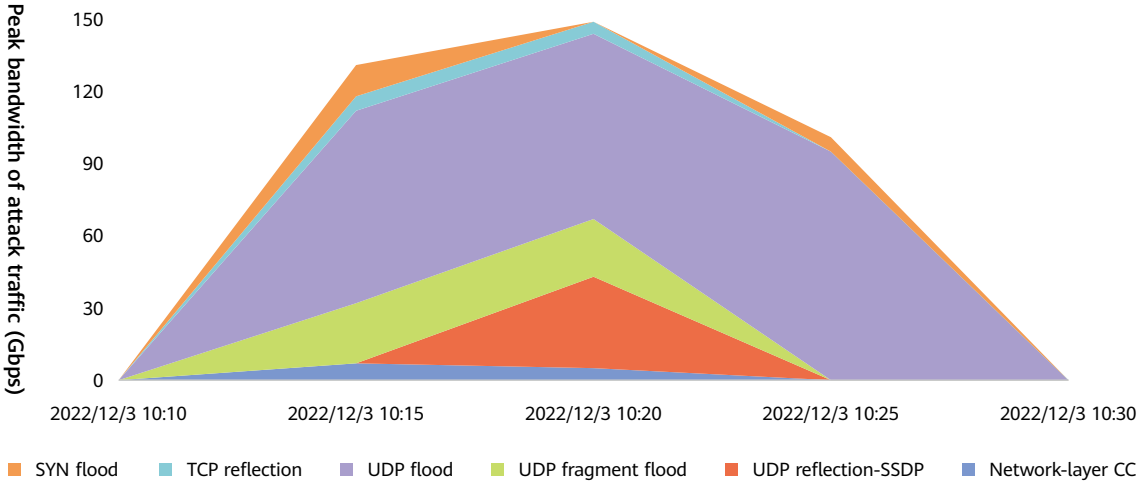
Traffic Distribution of the First Round of Attacks



Second round of attacks: attack complexity increases, challenging the defense success rate.

The second round of attacks started at 10:10 on December 3. The high-strength attacks lasted only 20 minutes, peaked at 149 Gbps. In addition to occupying bandwidth through UDP flood and UDP reflection attacks, attackers launched SYN flood and TCP reflection attacks besides network-layer CC attacks to challenge the defense success rate.

Traffic Distribution of the Second Round of Attacks

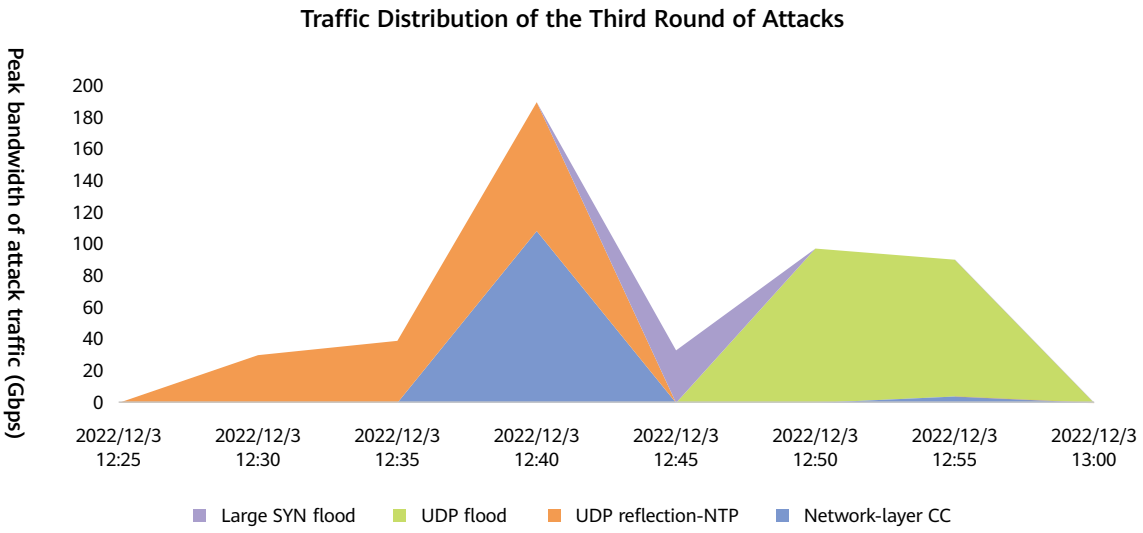


In this round of attacks, the attack tactics changed: In the category of network-layer CC attacks, the packet size of ACK flood attacks changed from large to small. In addition, to evade the Geo-IP policy, the attack source's location was switched to China, increasing the defense difficulty. In the category of UDP reflection attacks, SSDP reflection attacks took the place of NTP reflection attacks. At last, large UDP packets of more than 1500 bytes were used to generate a UDP fragment flood, further intensifying the attacks.

Third round of attacks: intensity of network-layer CC attacks is improved, and the large SYN flood is used to consume bandwidth.

The third round of attacks started at 12:25 on December 3, lasted for 35 minutes, and peaked at 189 Gbps.

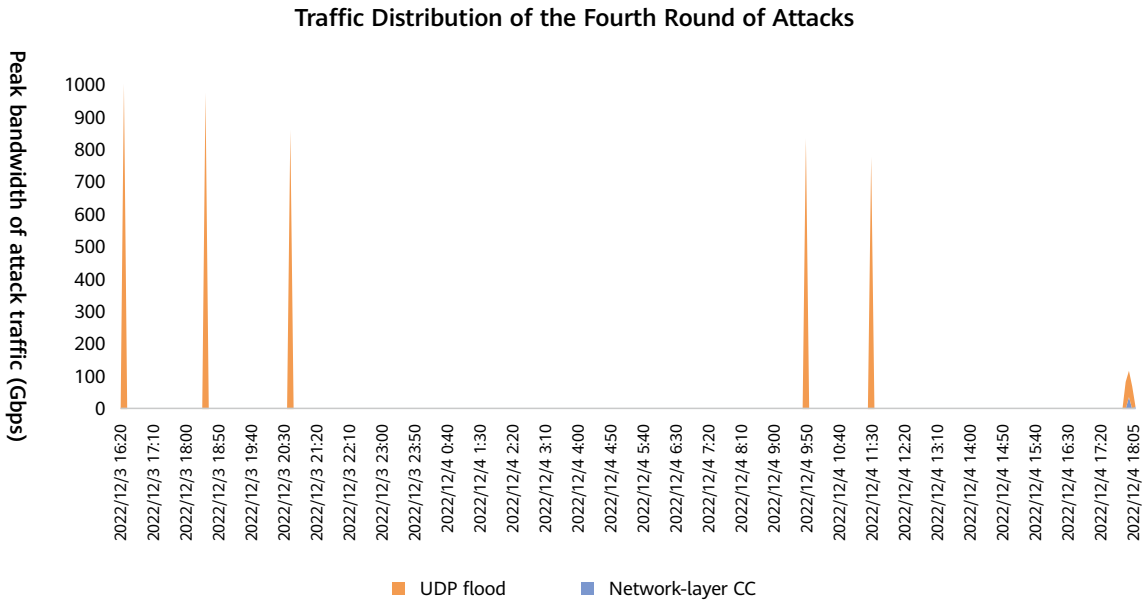
In this round of attacks, NTP reflection attacks replaced SSDP reflection attacks. In addition to UDP flood and UDP reflection attacks, large SYN flood attacks joined to overload bandwidth. Attackers continue to exploit botnets in China to launch network-layer CC attacks. The packet size of ACK flood attacks changed from small to large, further overloading bandwidth.



Fourth round of attacks: The attack intensity is increased to impact the network with hit-and-run attacks.

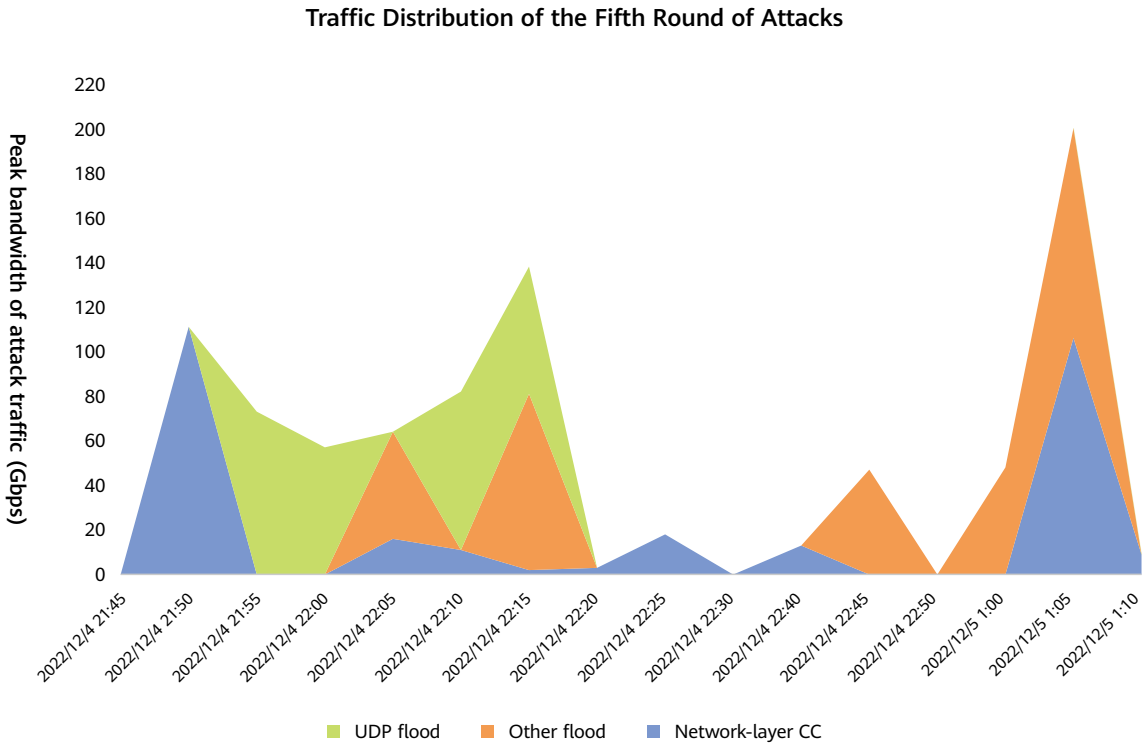
The first three rounds of attacks failed to break the defense. Then, the attacks entered the fourth round — high-strength UDP flood attacks were launched to occupy the network bandwidth again, increasing the defense cost.

This round of attacks started at 16:20 on December 3 and lasted 25 hours and 50 minutes. Traffic rate in this round had five peaks, which were 1002 Gbps, 974 Gbps, 860 Gbps, 834 Gbps, and 777 Gbps respectively. The interval between the first three flood peaks is 2 hours, and the interval between the last two flood peaks is also 2 hours, representing a typical hit-and-run attack. After this round of attacks, the attacked ISPs requested China Telecom to enable the policy of blocking UDP traffic on the upstream links, suppressing the UDP flood attack traffic.



Fifth round of attacks: Other flood attacks joined to threaten the bandwidth.

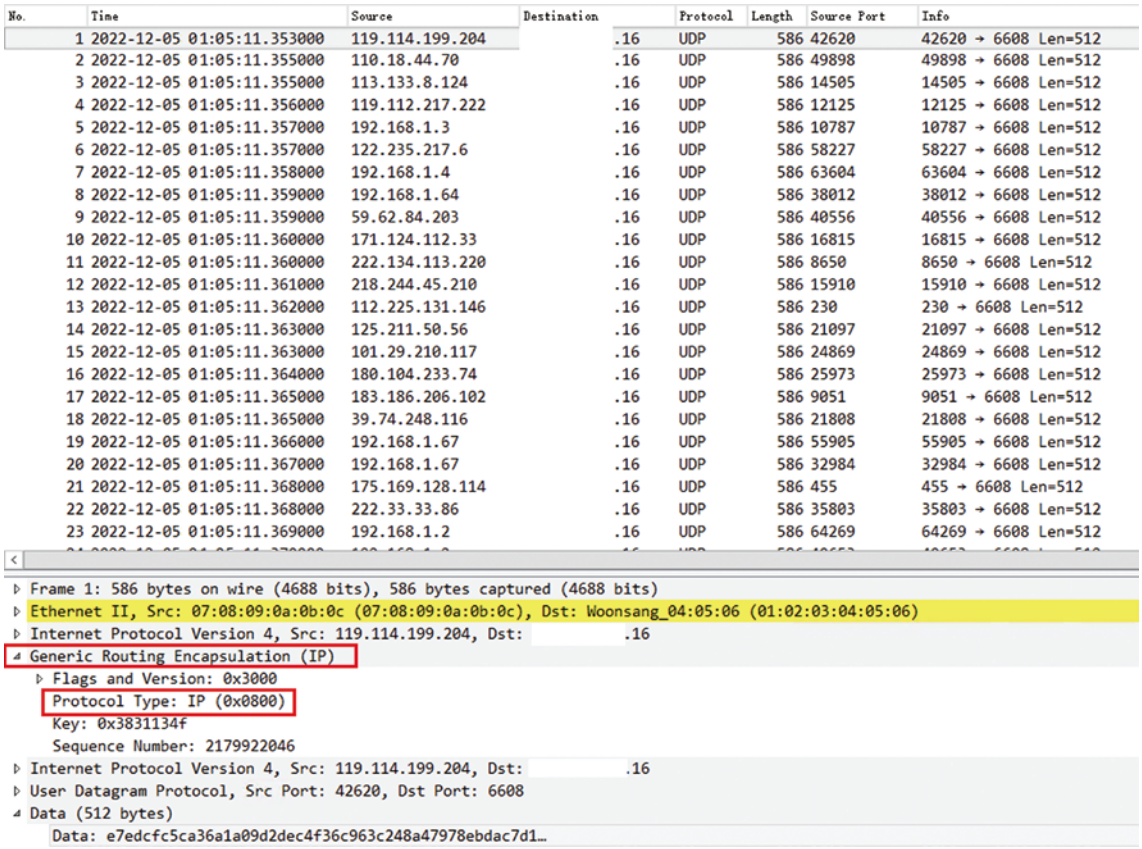
The fifth round of attacks started at 21:45 on December 4 and lasted for 3 hours and 25 minutes. In this round of attacks, the attacker transmitted UDP flood attack traffic through China Unicom and China Mobile links. To improve the attack effect, the attacker added other floods to increase the peak attack bandwidth to 200 Gbps. Moreover, the attacker continued to launch network-layer CC attacks using botnets in China to challenge the defense.



Other flood attacks, or other protocol flood attacks, use packets other than TCP, UDP, and ICMP packets.

Attack packets in the following figure show that the attack is a GRE flood.

Other Flood Attack Exploiting GRE



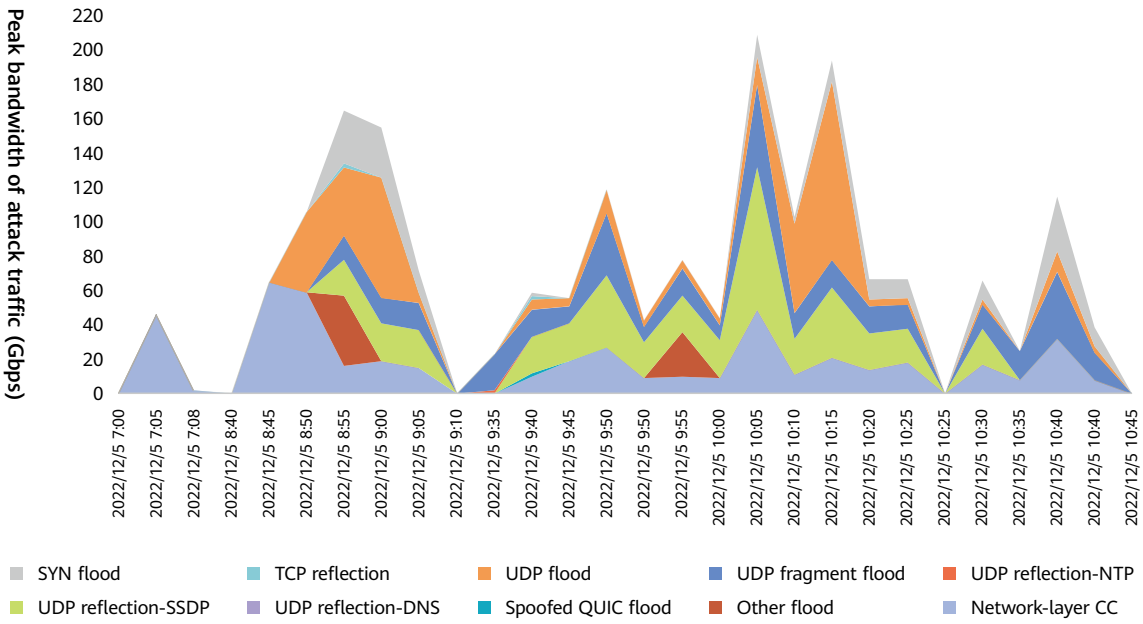
No.	Time	Source	Destination	Protocol	Length	Source Port	Info
1	2022-12-05 01:05:11.353000	119.114.199.204	.16	UDP	586	42620	42620 → 6608 Len=512
2	2022-12-05 01:05:11.355000	110.18.44.70	.16	UDP	586	49898	49898 → 6608 Len=512
3	2022-12-05 01:05:11.355000	113.133.8.124	.16	UDP	586	14505	14505 → 6608 Len=512
4	2022-12-05 01:05:11.356000	119.112.217.222	.16	UDP	586	12125	12125 → 6608 Len=512
5	2022-12-05 01:05:11.357000	192.168.1.3	.16	UDP	586	10787	10787 → 6608 Len=512
6	2022-12-05 01:05:11.357000	122.235.217.6	.16	UDP	586	58227	58227 → 6608 Len=512
7	2022-12-05 01:05:11.358000	192.168.1.4	.16	UDP	586	63604	63604 → 6608 Len=512
8	2022-12-05 01:05:11.359000	192.168.1.64	.16	UDP	586	38012	38012 → 6608 Len=512
9	2022-12-05 01:05:11.359000	59.62.84.203	.16	UDP	586	40556	40556 → 6608 Len=512
10	2022-12-05 01:05:11.360000	171.124.112.33	.16	UDP	586	16815	16815 → 6608 Len=512
11	2022-12-05 01:05:11.360000	222.134.113.220	.16	UDP	586	8650	8650 → 6608 Len=512
12	2022-12-05 01:05:11.361000	218.244.45.210	.16	UDP	586	15910	15910 → 6608 Len=512
13	2022-12-05 01:05:11.362000	112.225.131.146	.16	UDP	586	230	230 → 6608 Len=512
14	2022-12-05 01:05:11.363000	125.211.50.56	.16	UDP	586	21097	21097 → 6608 Len=512
15	2022-12-05 01:05:11.363000	101.29.210.117	.16	UDP	586	24869	24869 → 6608 Len=512
16	2022-12-05 01:05:11.364000	180.104.233.74	.16	UDP	586	25973	25973 → 6608 Len=512
17	2022-12-05 01:05:11.365000	183.186.206.102	.16	UDP	586	9051	9051 → 6608 Len=512
18	2022-12-05 01:05:11.365000	39.74.248.116	.16	UDP	586	21808	21808 → 6608 Len=512
19	2022-12-05 01:05:11.366000	192.168.1.67	.16	UDP	586	55905	55905 → 6608 Len=512
20	2022-12-05 01:05:11.367000	192.168.1.67	.16	UDP	586	32984	32984 → 6608 Len=512
21	2022-12-05 01:05:11.368000	175.169.128.114	.16	UDP	586	455	455 → 6608 Len=512
22	2022-12-05 01:05:11.368000	222.33.33.86	.16	UDP	586	35803	35803 → 6608 Len=512
23	2022-12-05 01:05:11.369000	192.168.1.2	.16	UDP	586	64269	64269 → 6608 Len=512

Sixth round of attacks: All attack vectors joined for violent attacks.

After five rounds of attack-defense confrontations, the attacks still cannot break through the defense. From 7:00 on December 5, the attacker used all available attack tactics for the last battle. This round of attacks lasted for 3 hours and 45 minutes. As some attack traffic was blocked by the UDP traffic-blocking policy enabled on the upstream China Telecom links, the attacks only peaked at 209 Gbps. In addition, the defense system deployed at the network border of the equipment room filtered out some attack traffic, suppressing the attack traffic within the threshold that the service system can carry.

This round of attacks was more sophisticated than ever. Attack tactics included spoofed QUIC flood, and multi-vector UDP reflection attacks (NTP, SSDP, and DNS reflection). In addition, the size of DNS reflection packets exceeded the maximum transmission unit (MTU), causing UDP fragment attacks and further intensifying attacks.

Traffic Distribution of the Sixth Round of Attacks

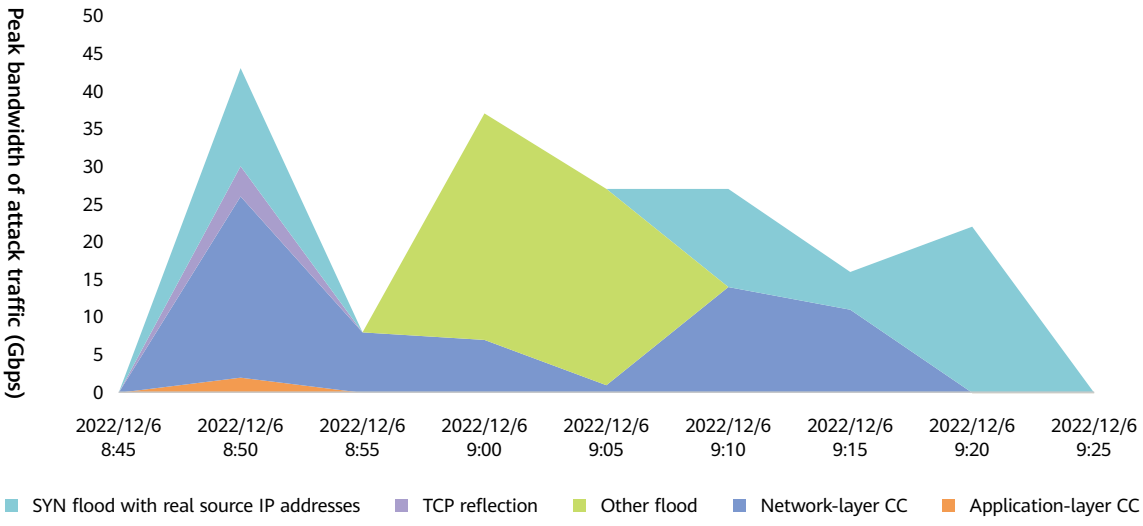


Seventh round of attacks: multi-vector CC attacks cannot break through the defense line, and the series of attacks end.

On December 6, the attacker used all types of network-layer attack tactics, and launched the last round of multi-vector botnet attacks, mainly SYN attacks with real source IP addresses, network-layer CC attacks, and HTTP flood attacks. The attacks lasted for 40 minutes and peaked at 43 Gbps.

In this round of attacks, SYN attacks with real source IP addresses and application-layer CC attacks were newly added. The application-layer CC attacks mainly used low-rate access requests to the root directory of the target server initiated by the GET method. Application-layer CC attacks peaked at 550 thousand rps, while SYN attacks peaked at 28.8 Mpps.

Traffic Distribution of the Seventh Round of Attacks



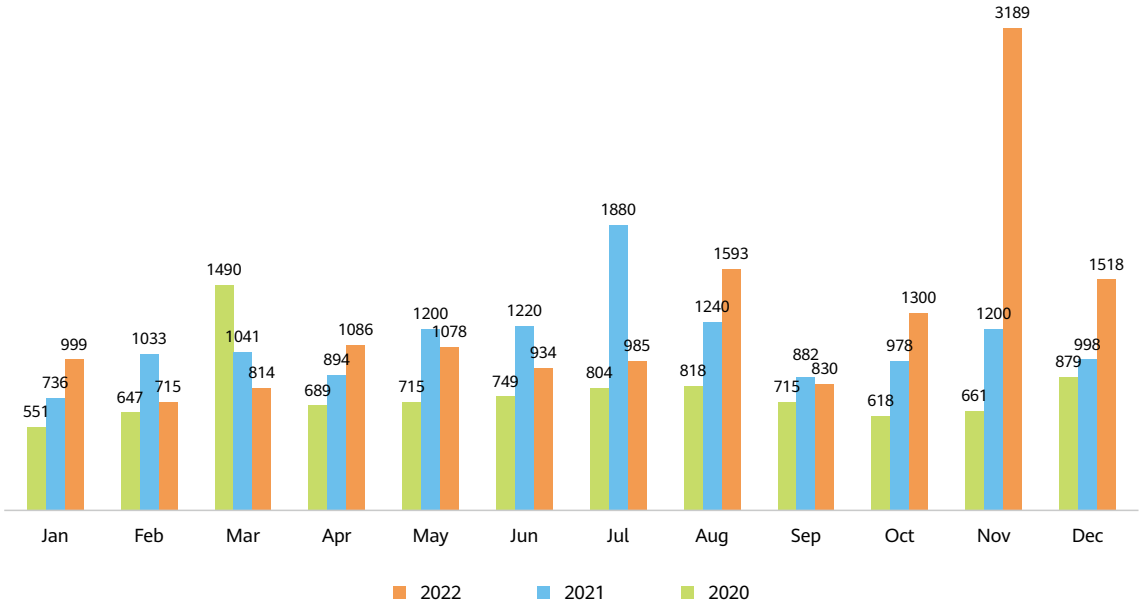
03 Situation and Trend

3.1 DDoS Attack Situation

3.1.1 Attack Intensity

Since 2022 Q2, terabit-strong attacks have become increasingly frequent. According to the China Telecom security team, in 2022, the largest bandwidth attack occurred at 8:08 on November 1. UDP flood was the main attack tactic. The attack peaked at 3.189 Tbps and lasted for 18 minutes. The victim IP address resided on the network of China Telecom Zhejiang.

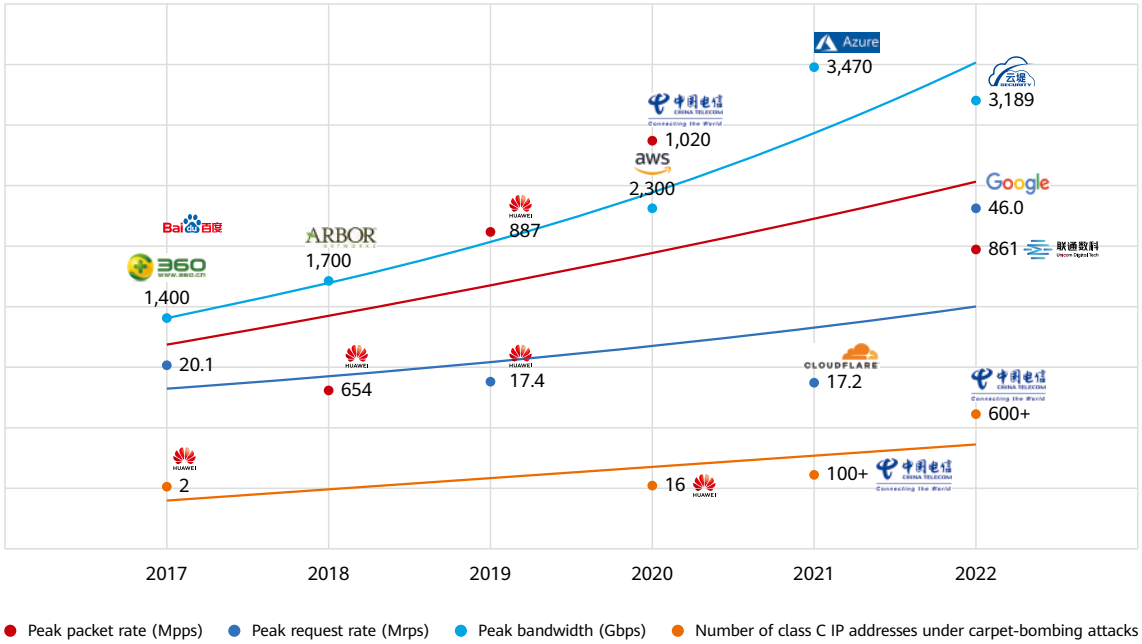
2020-2022 Peak Bandwidth Distribution of Attacks by Month (Gbps)



Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

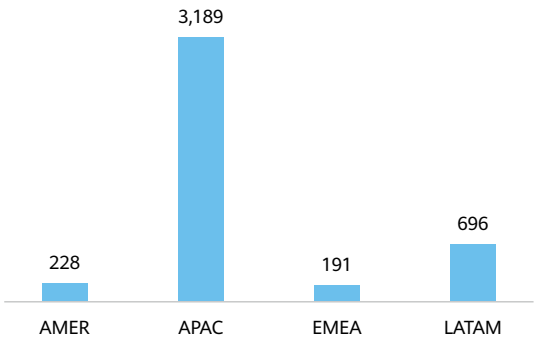
The costs involved in attacks continue to fall, meaning that the scale of attacks keeps growing. According to the annual strongest attacks recorded in recent years, the peak attack bandwidth, peak packet rate, peak rps (application-layer attacks), and carpet-bombing scale are on the rise.

Trend of DDoS Attacks with the Largest Bandwidth Globally



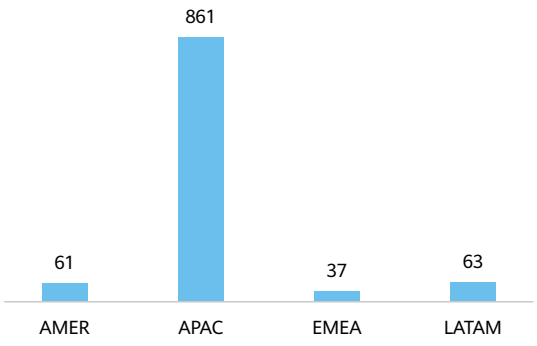
The attack with the largest bandwidth in the history of the Internet occurred in 2021, peaked at 3470 Gbps[2], and used multi-vector UDP reflection. The attack with the highest packet rate in the history of the Internet occurred in 2020, peaked at 1020 Mpps, and was a multi-vector attack (mainly using SYN flood and TCP reflection). Application-layer attacks at tens of millions of rps become the norm. The largest application-layer attack in the Internet history occurred in June 2022, peaked at 46 million rps[1]. The scale of carpet-bombing attacks rapidly increases, with the attacked class C IP address segments surging from 100+ in 2021 to 600+ in 2022. Ultra large-scale attacks pose severe challenges to defense costs.

2022 Peak Bandwidth Distribution of Attacks by Region (Gbps)



Source: China Telecom Cybersecurity, Nexusguard, Huawei

2022 Peak Packet Rate Distribution of Attacks by Region (Mpps)

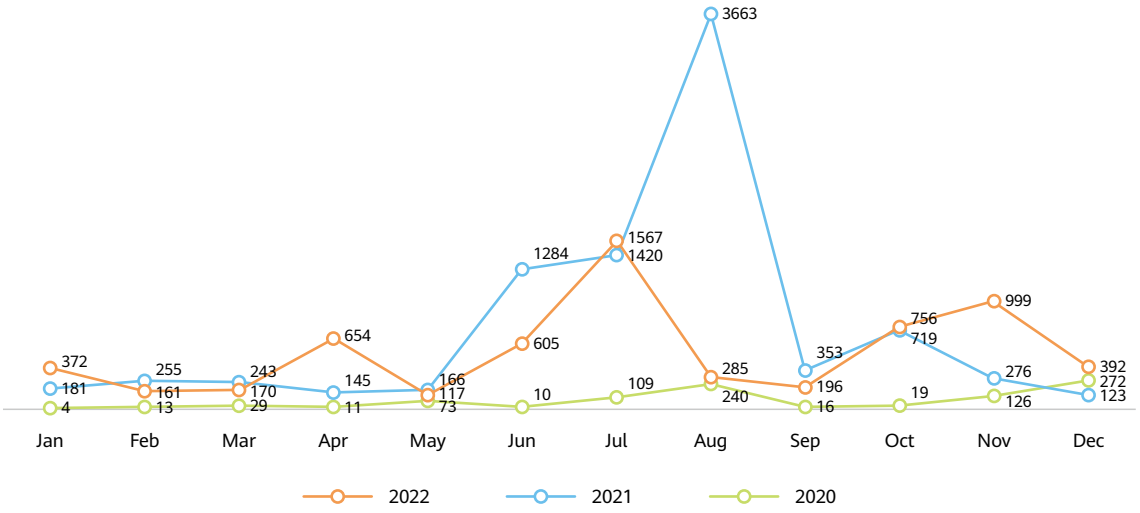


Source: China Telecom Cybersecurity, Nexusguard, Huawei

According to the preceding two figures, APAC suffered both the attacks with the largest peak bandwidth and those with the highest packet rate.

In 2022, attacks at over 500 Gbps occurred 6274 times, lower than that in 2021 (8828 times). July saw the largest number of attacks at over 500 Gbps, which is 1,567, accounting for 25% of the total attacks in the whole year.

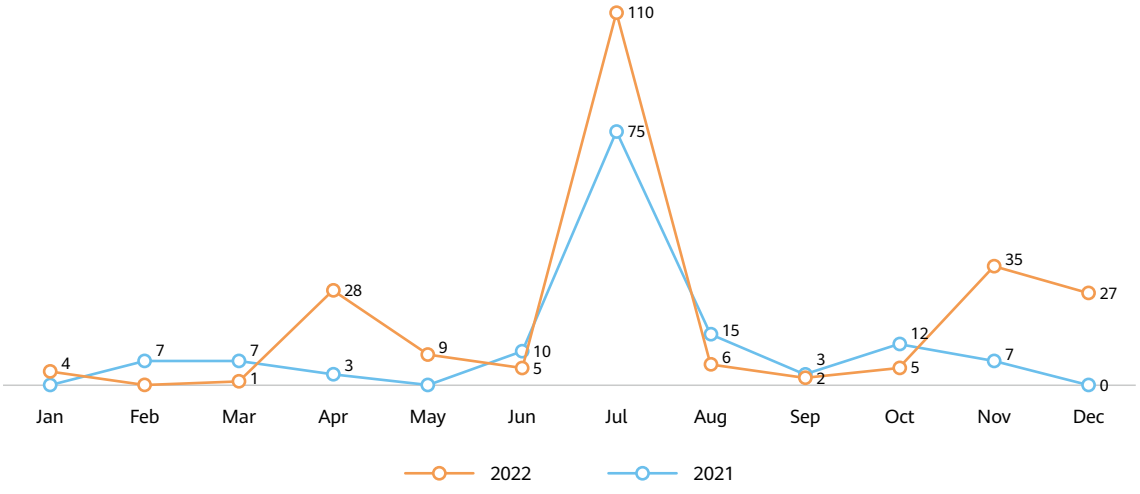
2020-2022 Distribution of Attacks at over 500 Gbps by Month



Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

Attacks at over 800 Gbps occurred 232 times in 2022, 1.67 times that in 2021. July also saw the largest number of attacks at over 800 Gbps, which is 110, accounting for 47% of the total attacks in the whole year.

2021-2022 Distribution of Attacks at over 800 Gbps by Month

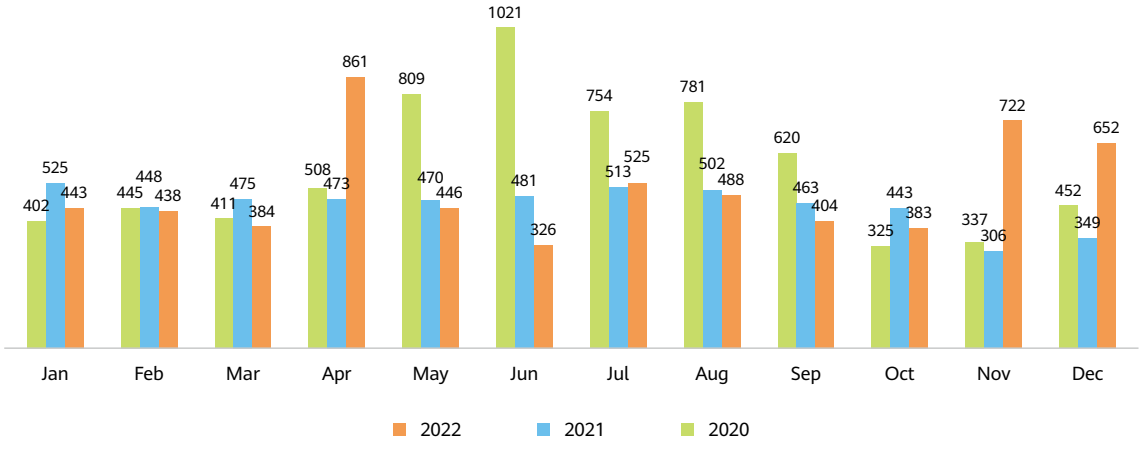


Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

According to the figures about monthly distribution of attacks at over 500 Gbps and over 800 Gbps, most of the ultra large-scale attacks happened in July and August, the summer vacation period, during which large-scale attacks targeting games are more active.

At 12:16 on April 9, 2022, the security team of China Telecom detected an attack with the maximum packet rate. The attack peaked at 861 Mpps, lasted for 11 minutes, and used the RST flood. The target IP address sourced from the network of China Telecom Zhejiang.

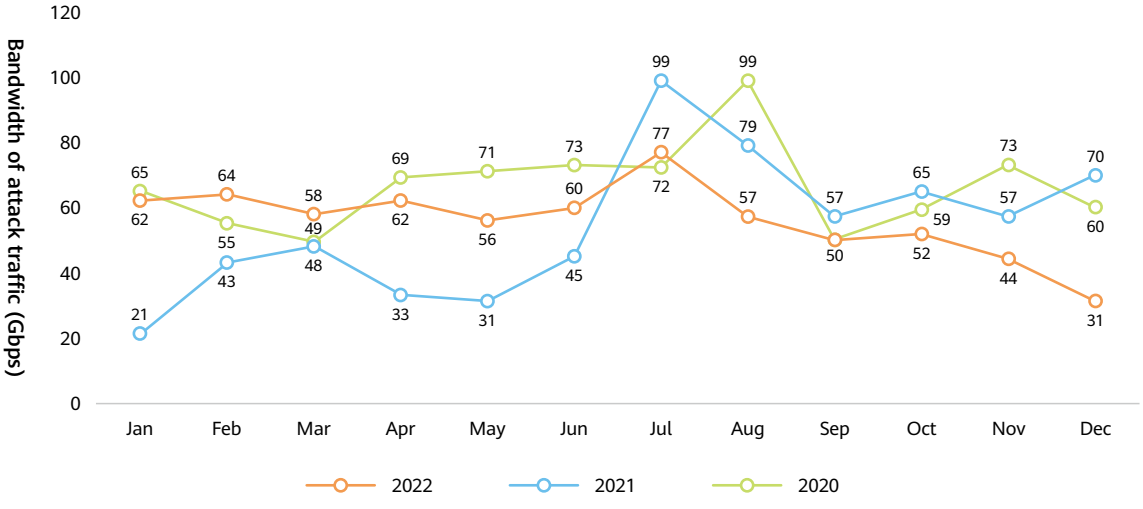
2020-2022 Peak Packet Rate Distribution of Attacks by Month (Mpps)



Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

The highest monthly average peak bandwidth of attacks in 2022 was 77 Gbps, which appeared in July. The annual average peak bandwidth of attacks in 2022 was 56 Gbps, slightly higher than 54 Gbps in 2021 and lower than 66 Gbps in 2020. In July and August each year, the average peak bandwidth of attacks is high. The main reason is that during the summer vacation period, large-scale attacks targeting games are more active.

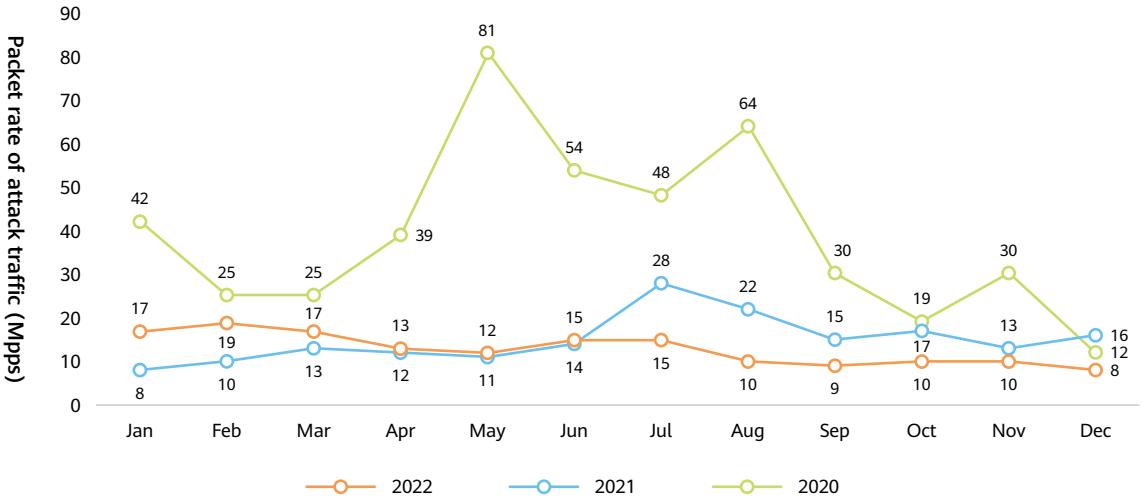
2020-2022 Average Peak Bandwidths of Attacks by Month



Source: China Telecom Cybersecurity, Huawei

The highest monthly average packet rate of attacks in 2022 was 19 Mpps, appeared in February. The annual average packet rate of attacks was 13 Mpps, which was close to 15 Mpps in 2021 and far lower than 39 Mpps in 2020.

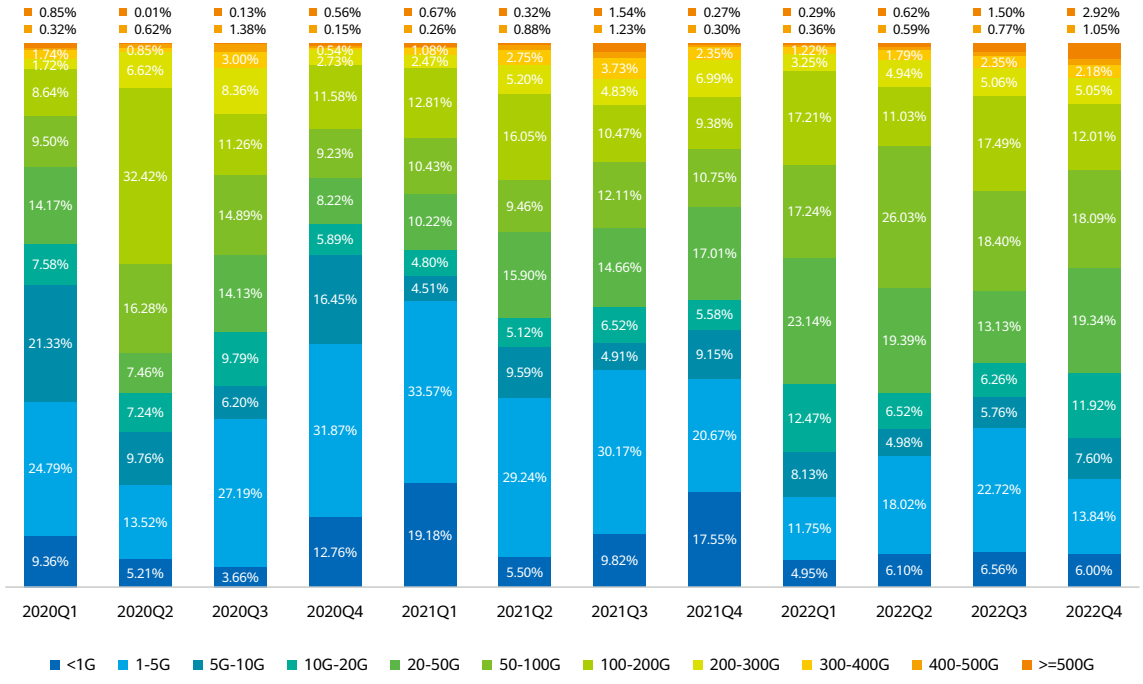
2020-2022 Average Peak Packet Rates of Attacks by Month



Source: China Telecom Cybersecurity, Huawei

The attack intensity trend in 2021 is similar to that in 2022. The average peak packet rate of attacks in 2020 is much higher than that in 2021 and 2022. The main reason is that in 2020, volumetric attacks (mainly using high-rate SYN flood and TCP reflection) were the mainstream, but in 2021 and 2022, UDP flood attacks with oversized packets and network-layer CC attacks (whose packet rate is comparatively low) became the norm.

2020-2022 Peak Bandwidth Ranges of Attacks

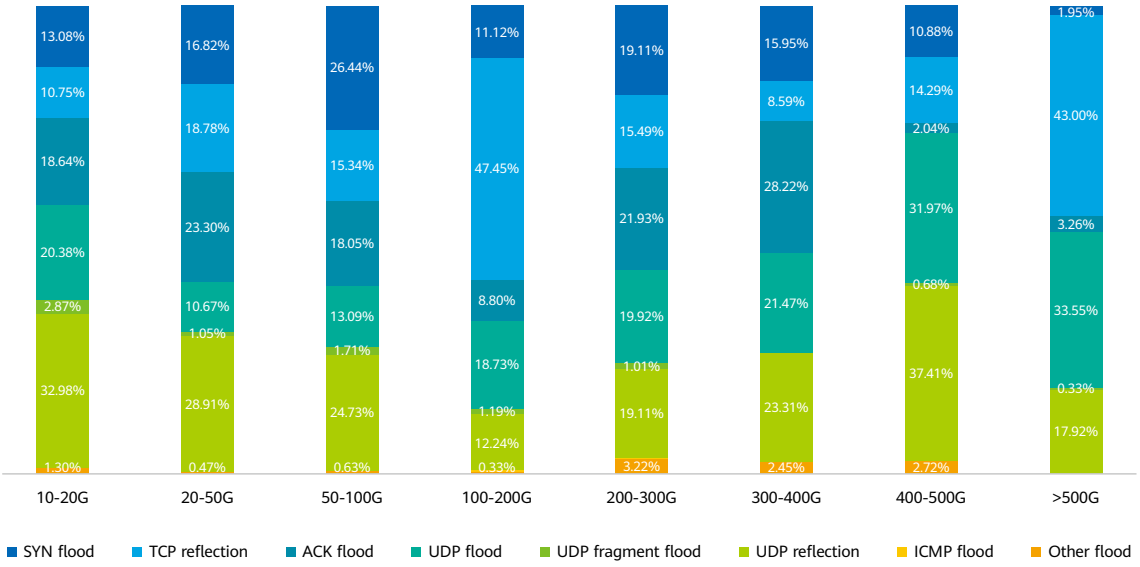


In 2022, Huawei detected 104,922 attacks exceeding 100 Gbps, with an average of 287 attacks per day. The number of attacks at over 100 Gbps is 1.5 times that of 2021 and 2.1 times that of 2020.

According to the distribution of peak attack traffic bandwidths, the number of attacks that peaked at 50-100 Gbps increased significantly in 2022. The quarterly average proportion was 19.94%, 1.86 times that in 2021 and 1.59 times that in 2020.

According to the distribution of peak attack traffic bandwidths of network-layer attacks in 2022, attacks peaked at 50-100 Gbps mainly consist of SYN flood, UDP reflection, ACK flood (network-layer CC), TCP reflection, and UDP flood attacks, accounting for 26.44%, 24.73%, 18.05%, 15.34% and 13.09% respectively.

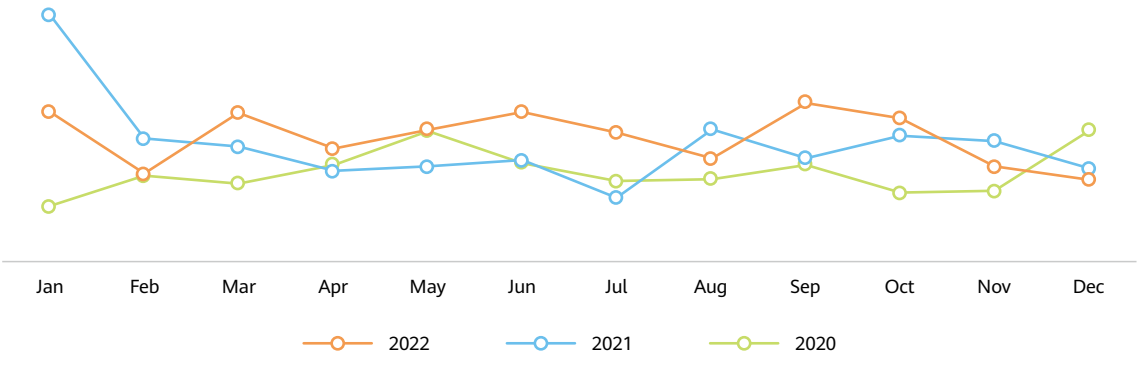
2022 Peak Bandwidth Ranges of Network-Layer Attacks



3.1.2 Attack Frequency

The frequency of DDoS attacks keeps increasing. The attack frequency in 2022 was 1.1 times that in 2021 and 1.4 times that in 2020.

2020-2022 Distribution of Network-Layer Attacks by Month

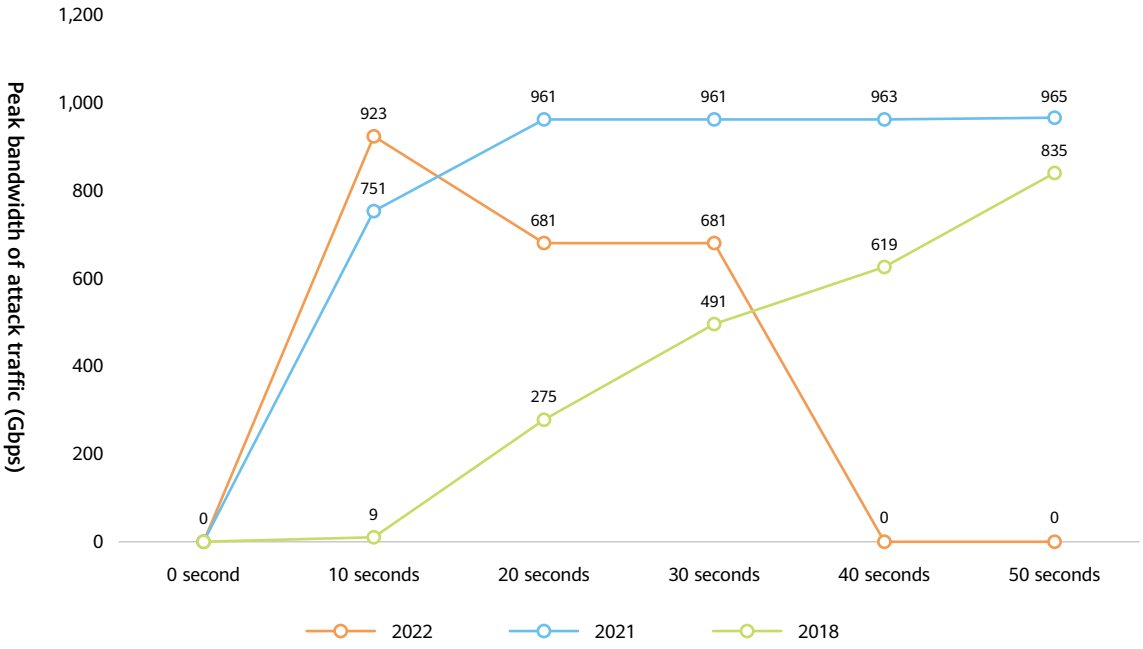


Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

3.1.3 Attack Speed

Volumetric attacks surge in seconds. According to the attack traffic ramp-up speed statistics in 2018, 2021, and 2022, the attack traffic ramp-up speeds keep increasing. In 2018, it takes 50 seconds for the peak attack traffic to ramp up to the range of 800 Gbps to 1 Tbps; however, it takes 20 seconds in 2021 and only 10 seconds in 2022.

2018-2022 Traffic Ramp-Up Speeds of Attacks



3.1.4 Attack Complexity

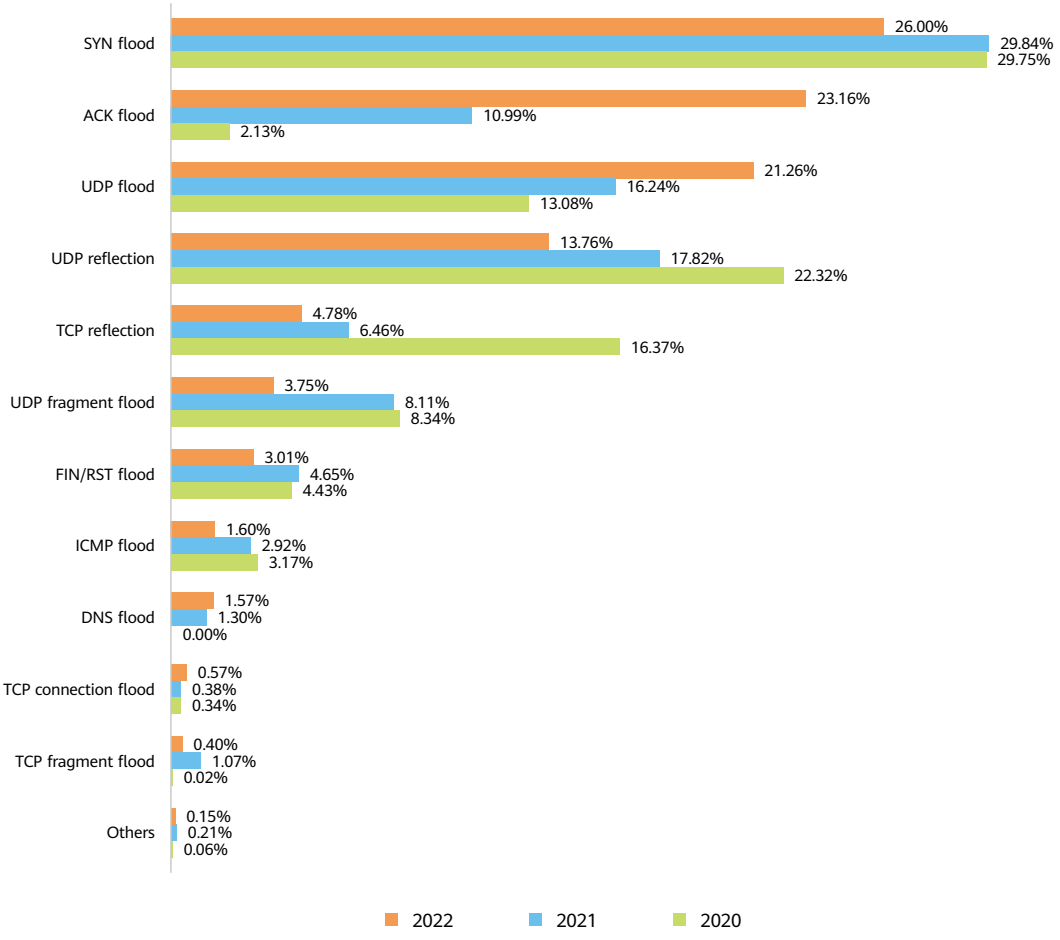
1. Distribution of network-layer attacks by type

In 2022, SYN flood, ACK flood, UDP flood, UDP reflection, and TCP reflection attacks are top 5 network-layer attacks. In the past three years, the proportions of ACK flood and UDP flood attacks have been on the rise year by year.

In 2022, ACK flood attacks accounted for 23.16%, 2.1 times that in 2021 and 10.9 times that in 2020. The main reason why the number of ACK flood attacks increases rapidly is that network-layer CC attacks launched by the Mirai botnet are destructive, and hard to be defended against. As attack costs keep decreasing, network-layer CC attacks have become the trump card for attacking TCP servers.

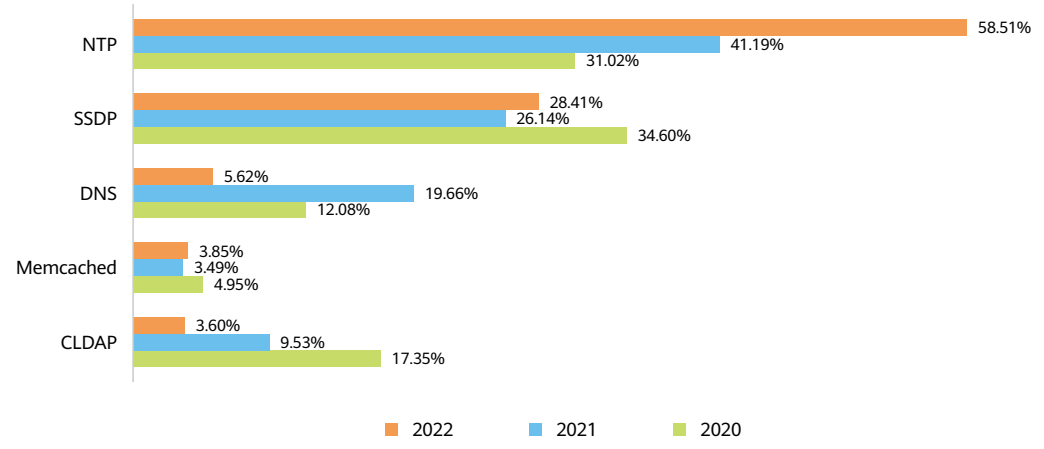
In 2022, UDP flood attacks accounted for 21.26%, 1.3 times that in 2021 and 1.6 times that in 2020. The rapid growth of UDP flood attacks is directly related to the release of the standardized QUIC protocol in May 2021. After the release, spoofed QUIC flood attacks (a type of UDP flood targeting port 443) became active. Moreover, the packet sending rate of the attack platforms increases, and the cost of terabit-strong UDP flood attacks is low. As a result, UDP flood attacks with oversized packets targeting TCP servers are more active year by year, increasing the proportion of UDP flood attacks.

2020-2022 Types of Network-Layer Attacks



Among the top 5 UDP reflection attacks, NTP reflection attacks increased significantly, accounting for 58.51% in 2022, compared to 41.19% in 2021.

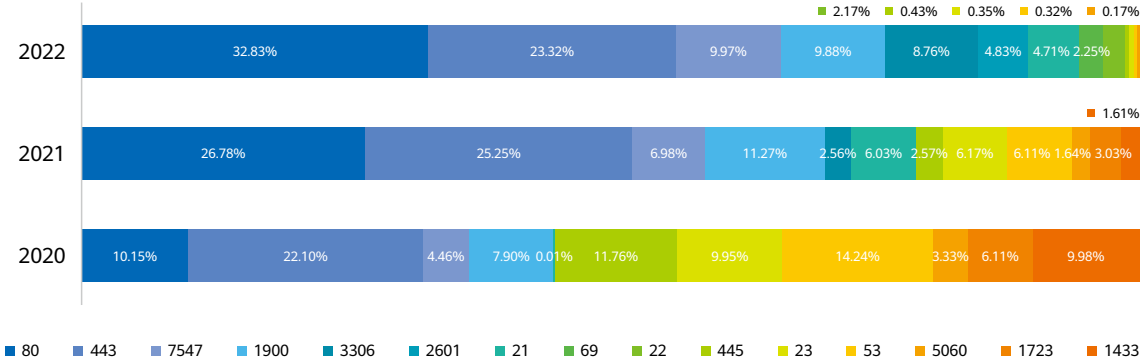
2020-2022 Top 5 UDP Reflection Attack Types



Source: China Unicom Digital Tech, Huawei

In 2022, 80 (HTTP), 443 (HTTPS), 7547 (CWMP), 1900 (SSDP) and 3306 (MySQL) were the top 5 TCP reflection source ports. In particular, traffic from ports 80 and 433 accounted for 56.16% of the total TCP reflection traffic, challenging the effectiveness of traditional static filtering policies on IDCs and cloud DCs. To reduce the impact of defense on services, the TCP reflection-filtering policy must be automatically created when an attack occurs and automatically canceled when the attack terminates.

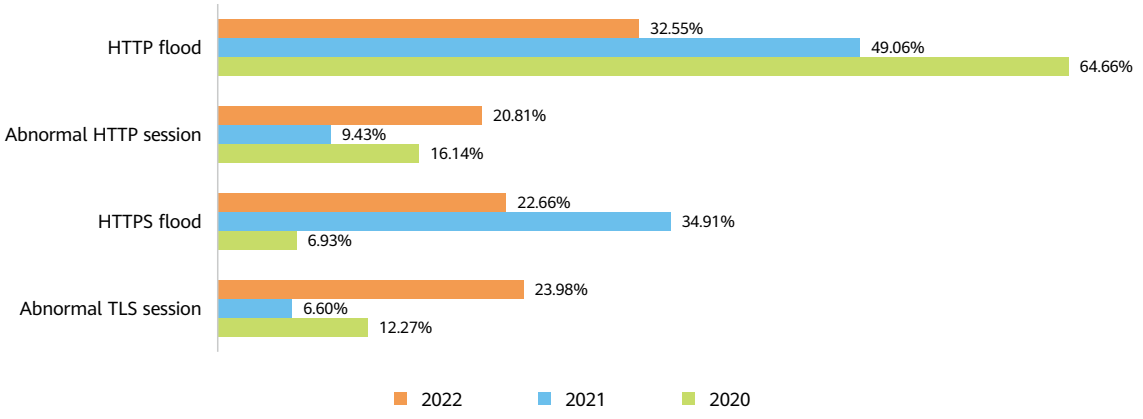
2020-2022 Source Ports of Typical TCP Reflection Attacks



2. Distribution of application-layer attacks by type

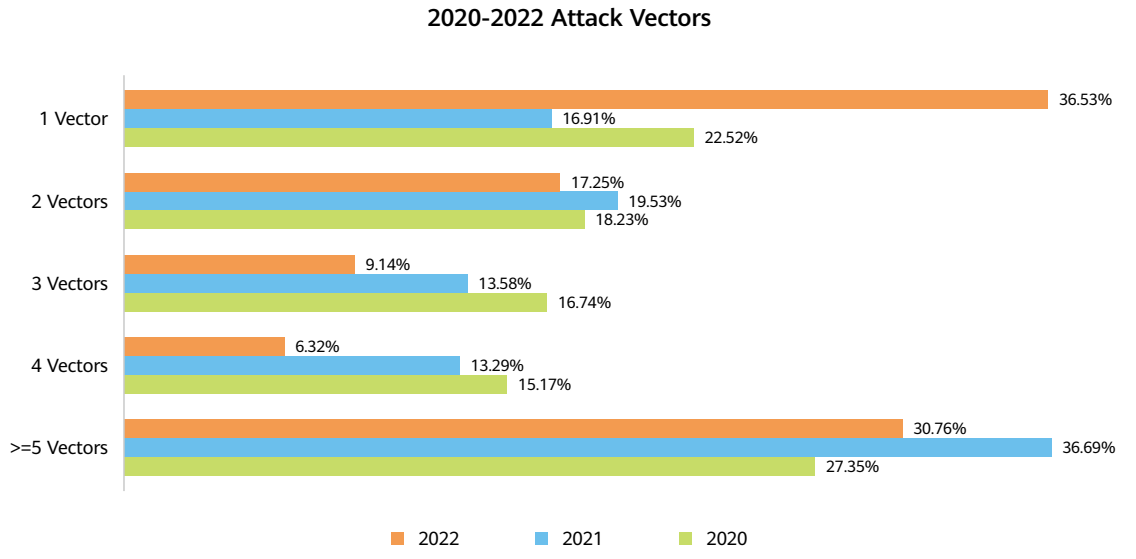
In 2022, as HTTP traffic is gradually encrypted using TLS, the proportion of HTTP flood attacks decreased. HTTP and HTTPS ports are vulnerable to network-layer CC attacks with obvious effectiveness. As a result, the number of HTTP and TLS abnormal session attacks increases. In addition, TLS attacks are more active because they are easy to launch and can significantly decrease the performance of the target servers. Due to the preceding factors, the proportion of TLS abnormal session attacks to the total number of attacks increased in 2022.

2020-2022 Application-Layer Attack Types



3. Attack vector distribution

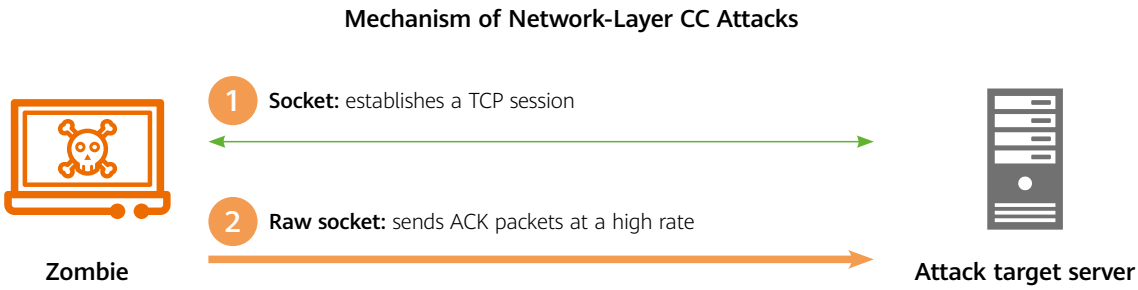
In 2022, the proportion of multi-vector attacks accounted for 63.47%, which is less than that in 2021. However, multi-vector attacks were still the mainstream.



Source: China Unicom Digital Tech, Huawei

4. Network-layer CC attacks continuously evolving with intensified destructiveness and enhanced defense evasion capabilities

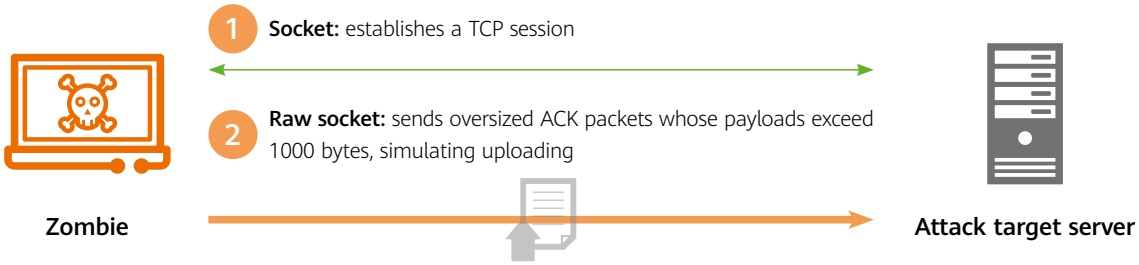
In the past two years, network-layer CC attacks have become one of the most difficult attacks to defend against.



Network-layer CC attacks combine the techniques used in session attacks and those in flood attacks with forged source IP addresses. Base on the analysis result of the DDoS attack code of the Mirai botnet, the basic tactic of a network-layer CC attack is found: An attacker establishes a TCP session with a target server through a socket. After obtaining session information, the attacker sends ACK packets to the target server at high speed through a raw socket. According to attack data obtained by Huawei, the maximum packet sending rate of a single attack source can reach 200,000 pps, exhausting the target server performance instantly.

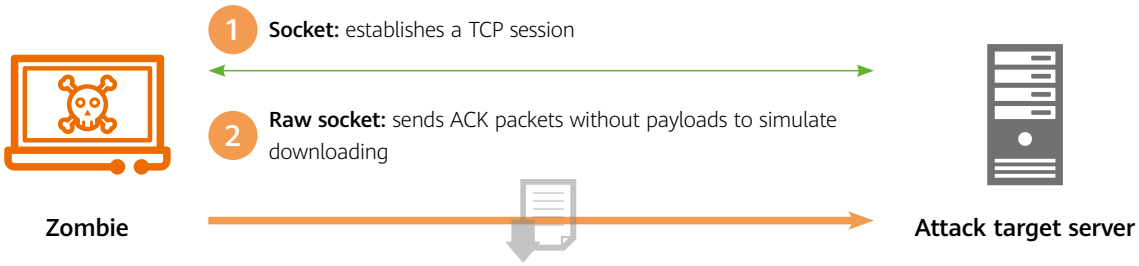
To improve the attack effect and evade defense, network-layer CC attacks evolve into three attack patterns. The first is to send oversized ACK packets whose payloads exceed 1000 bytes, simulating uploading. In this attack pattern, the most prominent feature is using ultra-large bandwidth traffic to quickly congest a network. The attack traffic surges in seconds and can ramp up to terabit-level bps within 20 seconds. The maximum peak bandwidth of the network-layer CC attacks detected by Huawei is 912 Gbps, while the peak packet rate is only 79 Mpps.

Mechanism of Network-Layer CC Attacks Simulating Uploading



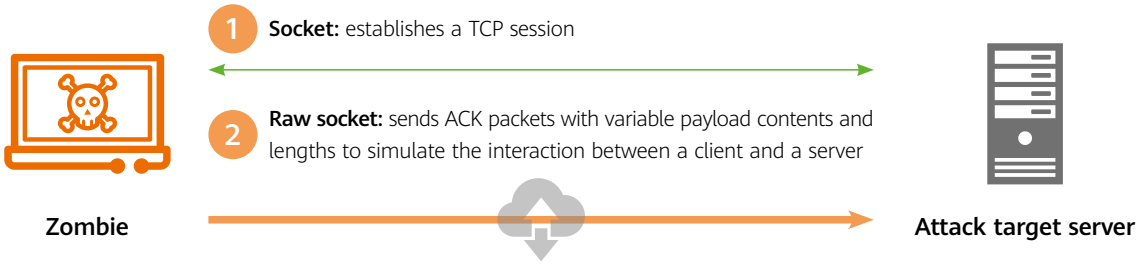
The second is to send ACK packets without payloads to simulate downloading. The maximum peak packet rate of network-layer CC attacks detected by Huawei is 139 Mpps, while the corresponding peak attack bandwidth is only 89 Gbps.

Mechanism of Network-Layer CC Attacks Simulating Downloading



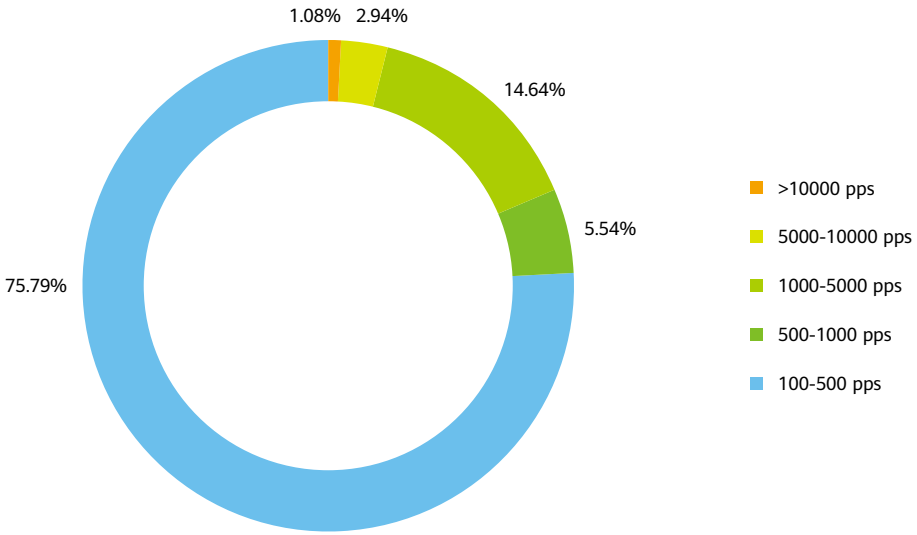
The third is to send ACK packets with variable payload contents and lengths to simulate the interaction between a client and a server.

Mechanism of Network-Layer CC Attacks Simulating the Interaction Between a Client and a Server



To increase the defense difficulty, an attacker sends packets in high-rate mode and low-rate mode alternately in an attack.

Packet Sending Rate of Zombies in a Network-Layer CC Attack

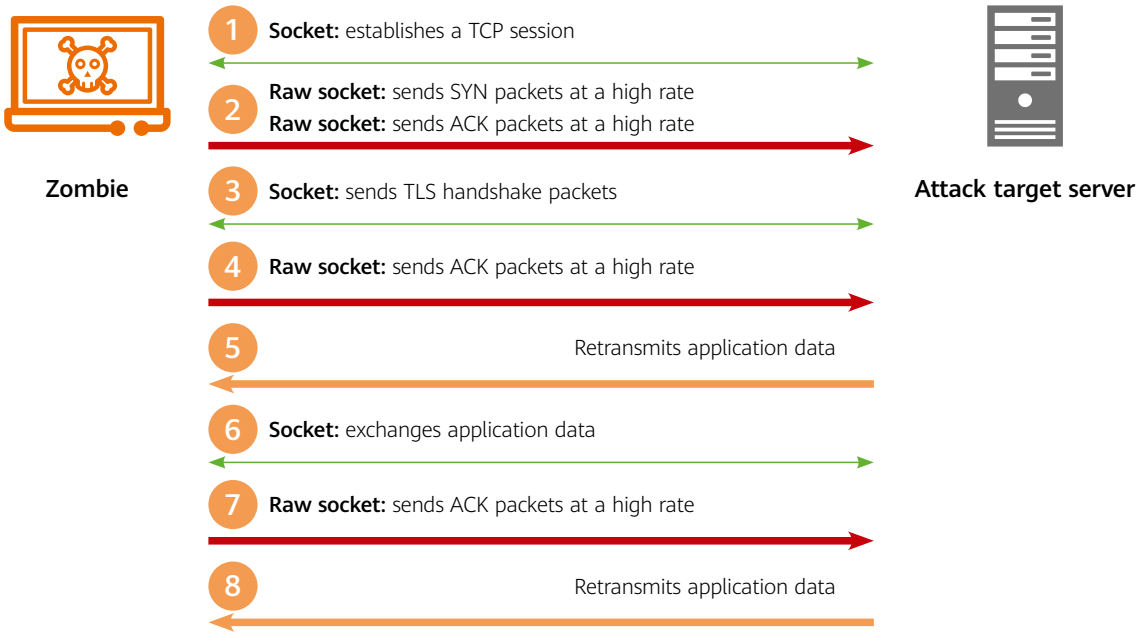


The average packet sending rate of zombies in high-rate mode ranges from 1000 pps to 5000 pps. Some high-performance zombies send packets at a rate of tens of thousands or even hundreds of thousands. The average packet sending rate of zombies in medium-rate mode ranges from 500 pps to 1000 pps. The average packet sending rate of zombies in low-rate mode ranges from 100 pps to 500 pps, which is close to the access rate of legitimate service clients, making the identification and defense more difficult.

From 2021 to 2022, network-layer CC attacks develop multiple variants, posing challenges to the effectiveness of defense technologies.

» **Variant 1: Attacks that carry complete HTTPS interaction sessions, triggering the fast retransmission mechanism of the target server**

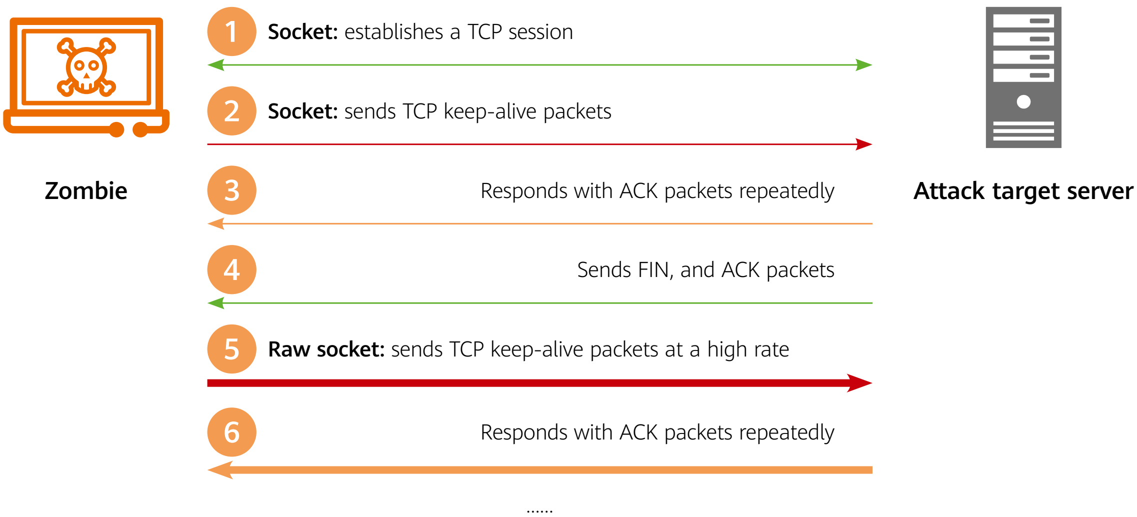
Mechanism of Attacks that Carry Complete HTTPS Interaction Sessions



A complete TLS session is established between an attacking device and a target server. During application data exchange, SYN packets and ACK packets are replayed at a high rate. The replayed ACK packets trigger the fast retransmission mechanism of the target server, and multiple retransmissions of the server form an outbound bandwidth attack.

» **Variant 2: Attacks that use forged TCP keep-alive packets to maintain TCP sessions for a long time**

Mechanism of Attacks that Use Forged TCP Keep-Alive Packets to Maintain TCP Sessions



After establishing a TCP connection with a target server using a socket, the attacking device forges a TCP keep-alive packet based on the ACK packet received after the TCP three-way handshake, and send it to the target server so that the server responds with another ACK packet. Then, the attacking device sends keep-alive packets at a high rate through a raw socket. As a result, the TCP service of the server is occupied for a long time.

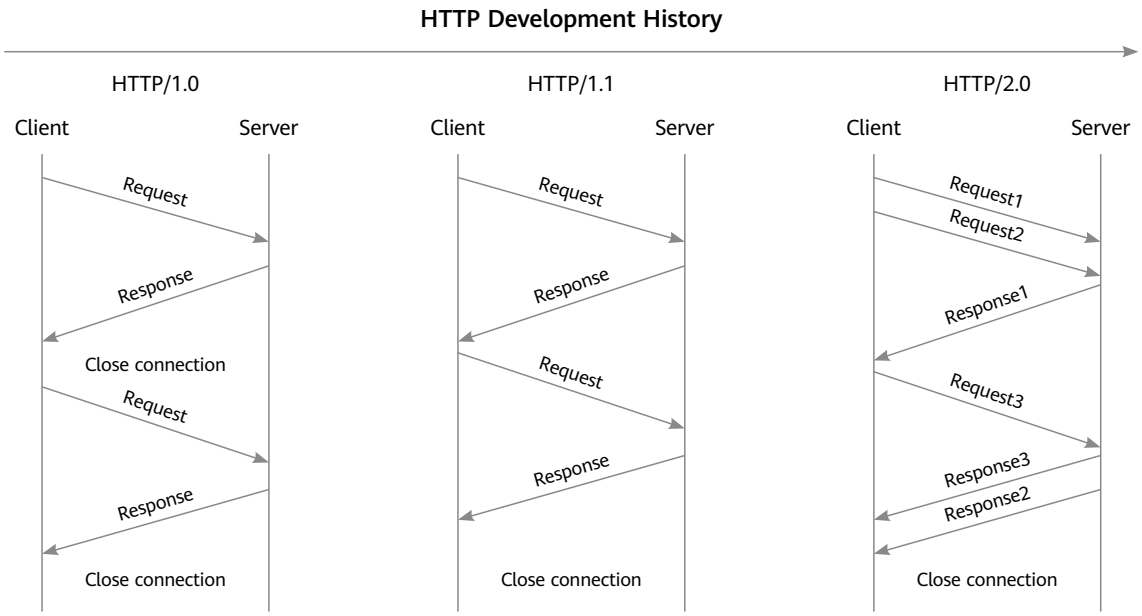
Obtained Packets of Attacks that Use Forged TCP Keep-Alive Packets to Maintain TCP Sessions

Time	Source	Protocol	Destination	Length	Sport	Info
2023-01-03 08:41:16.110441	39.156.74.251	TCP	.154	74	22849	22849 → 114 [SYN] Seq=0 Win=8192 Len=0 MSS=1200 SACK_PERM=1 WS=128
2023-01-03 08:41:16.110465	.154	TCP	39.156.74.251	70	114	114 → 22849 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1
2023-01-03 08:41:16.110581	39.156.74.251	TCP	.154	74	22849	[TCP Out-Of-Order] 22849 → 114 [SYN] Seq=0 Win=8192 Len=0 MSS=1200 SACK_PERM=1 WS=128
2023-01-03 08:41:16.110581	39.156.74.251	TCP	.154	74	22849	[TCP Out-Of-Order] 22849 → 114 [SYN] Seq=0 Win=8192 Len=0 MSS=1200 SACK_PERM=1 WS=128
2023-01-03 08:41:16.110597	.154	TCP	39.156.74.251	70	114	[TCP Out-Of-Order] 114 → 22849 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1
2023-01-03 08:41:16.110602	.154	TCP	39.156.74.251	70	114	[TCP Out-Of-Order] 114 → 22849 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1
2023-01-03 08:41:16.114430	39.156.74.251	TCP	.154	68	22849	22849 → 114 [ACK] Seq=1 Ack=1 Win=1868800 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114430	39.156.74.251	TCP	.154	68	22849	22849 → 114 [PSH, ACK] Seq=1 Ack=1 Win=7680000 Len=1[Packet size limited during capture]
2023-01-03 08:41:16.114437	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=1 Win=1868800 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114437	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=1 Win=1868800 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114437	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=1 Win=1868800 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114452	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=1 Win=1868800 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114453	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [PSH, ACK] Seq=1 Ack=1 Win=7680000 Len=1[Packet size limited during capture]
2023-01-03 08:41:16.114468	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [PSH, ACK] Seq=1 Ack=1 Win=7680000 Len=1[Packet size limited during capture]
2023-01-03 08:41:16.114469	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [PSH, ACK] Seq=1 Ack=1 Win=7680000 Len=1[Packet size limited during capture]
2023-01-03 08:41:16.114469	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [PSH, ACK] Seq=1 Ack=1 Win=7680000 Len=1[Packet size limited during capture]
2023-01-03 08:41:16.114528	.154	TCP	39.156.74.251	68	114	114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114532	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114537	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114541	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114547	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:16.114554	.154	TCP	39.156.74.251	74	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0 SLE=1 SRE=2
2023-01-03 08:41:16.114561	.154	TCP	39.156.74.251	74	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0 SLE=1 SRE=2
2023-01-03 08:41:16.114565	.154	TCP	39.156.74.251	74	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0 SLE=1 SRE=2
2023-01-03 08:41:16.114578	.154	TCP	39.156.74.251	74	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=1 Ack=2 Win=14600 Len=0 SLE=1 SRE=2
2023-01-03 08:41:46.335284	.154	TCP	39.156.74.251	68	114	114 → 22849 [FIN, ACK] Seq=1 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360238	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=2855261933 Win=131328 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360259	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=2 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360309	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=2855261933 Win=131328 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360324	.154	TCP	39.156.74.251	68	114	[TCP Keep-Alive ACK] 114 → 22849 [ACK] Seq=2 Ack=2 Win=14600 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360388	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=2855261933 Win=131328 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360388	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=2855261933 Win=131328 Len=0[Packet size limited during capture]
2023-01-03 08:41:46.360388	39.156.74.251	TCP	.154	68	22849	[TCP Keep-Alive] 22849 → 114 [ACK] Seq=1 Ack=2855261933 Win=131328 Len=0[Packet size limited during capture]

Note: Packets are obtained between the NAT gateway and the attack target server, and the attack target port is port 80. The NAT gateway maps port 80 to port 114 (shown in the preceding packet obtaining diagram). Moreover, the flow mirroring software on the customer network adds 10-byte payloads to packets. As a result, Packet size limited during capture is displayed multiple times when Wireshark parses the packet obtaining file.

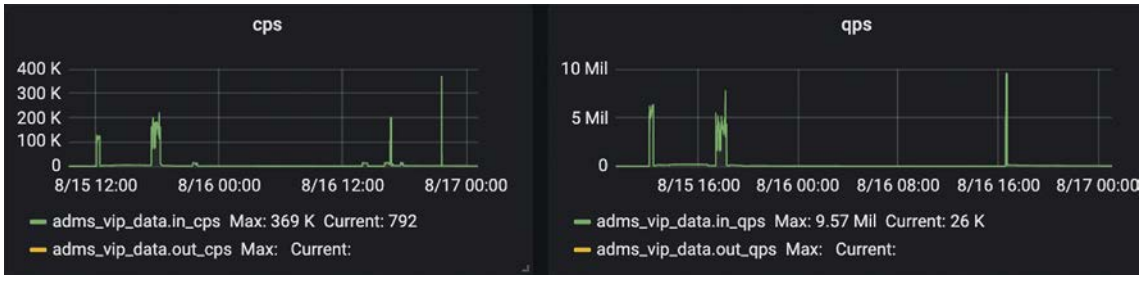
5. HTTP/2 multiplexing being abused to launch application-layer CC attacks at tens of millions of rps

HTTP/2 uses a more efficient transmission mechanism that bases on binary data frames, instead of texts used by HTTP 1.x. In this way, HTTP/2 does not rely on TCP connections to implement multi-flow parallel transmission. That is, all requests and responses under the same domain name can be concurrently transmitted over a single TCP connection. This binary flow-based transmission mechanism is also called multiplexing. Multiplexing transmission enables the transmission of parallel requests under the same domain name and improves the transmission performance of HTTP.



However, attackers abuse the HTTP/2 multiplexing feature. They send a large number of HTTP/HTTPS requests in a TCP connection to launch large-scale application-layer CC attacks without exploiting botnets. As early as October 2019, security personnel disclosed that the HTTP/2 multiplexing feature may be exploited by attackers to launch Multiplexed Asymmetric Attacks[3]. In August 2022, Baidu security team detected multiple application-layer CC attacks at tens of millions rps using the HTTP/2 multiplexing feature. During the attacks, a single zombie sent more than 500 HTTP requests in a TCP connection.

The following figure is the timing diagram of an attack. During the attack, the maximum session creation rate is 369,000 connections per second (cps), the maximum request rate is 9,570,000 queries per second (qps), and the ratio of the session creation rate to the request rate is 1:26. However, under normal conditions, the ratio is between 1:1 and 1:5. Based on the ratio of the session creation rate to the request rate, you can determine whether a Multiplexed Asymmetric Attack occurs.

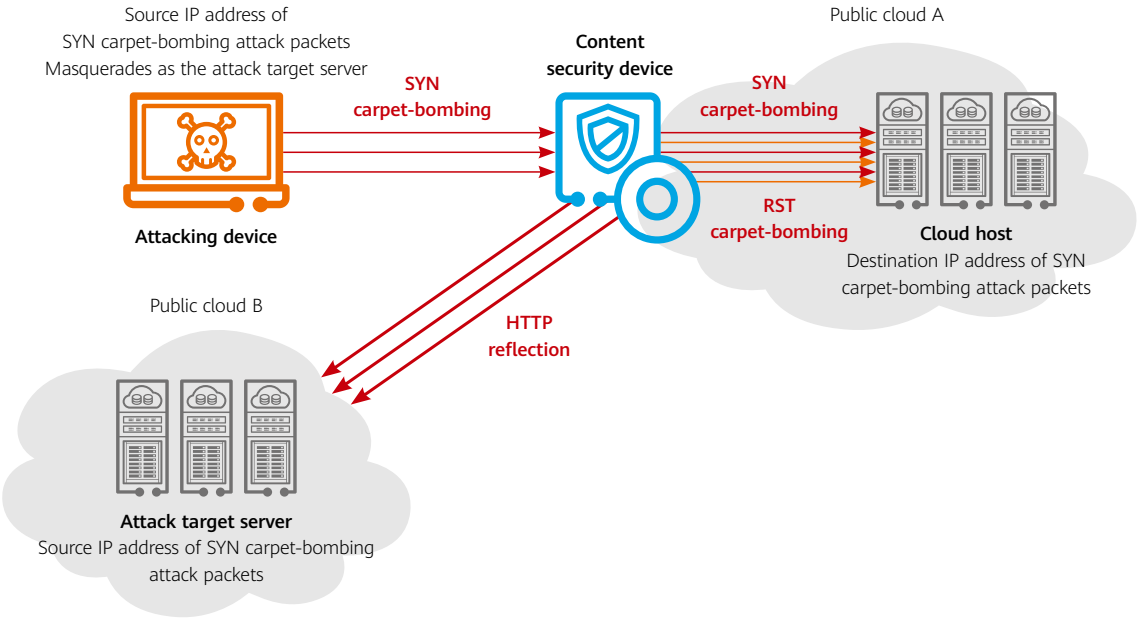


6. SYN carpet-bombing attacks triggering HTTP reflection and RST carpet-bombing attacks by exploiting vulnerabilities of content security devices

A public cloud has abundant IP address resources. As a result, the public cloud not only easily falls victim to carpet-bombing and reflection attacks, but also passively becomes the source of these attacks. In 2021, public clouds suffered SYN carpet-bombing attacks, and passively became the source of TCP carpet-bombing and reflection attacks. In 2022, carpet-bombing attacks targeting public clouds further evolved, forming a new attack tactic that combines carpet-bombing and HTTP reflection.

In May 2022, vulnerabilities of content security devices at the borders of multiple large-scale public clouds were exploited. Attackers used SYN packets with payloads carrying illegal content to launch carpet-bombing attacks on public clouds, and such attacks further triggered HTTP reflection and RST carpet-bombing attacks. The following figure shows the mechanism of HTTP reflection attacks initiated using a content security device at the public cloud border.

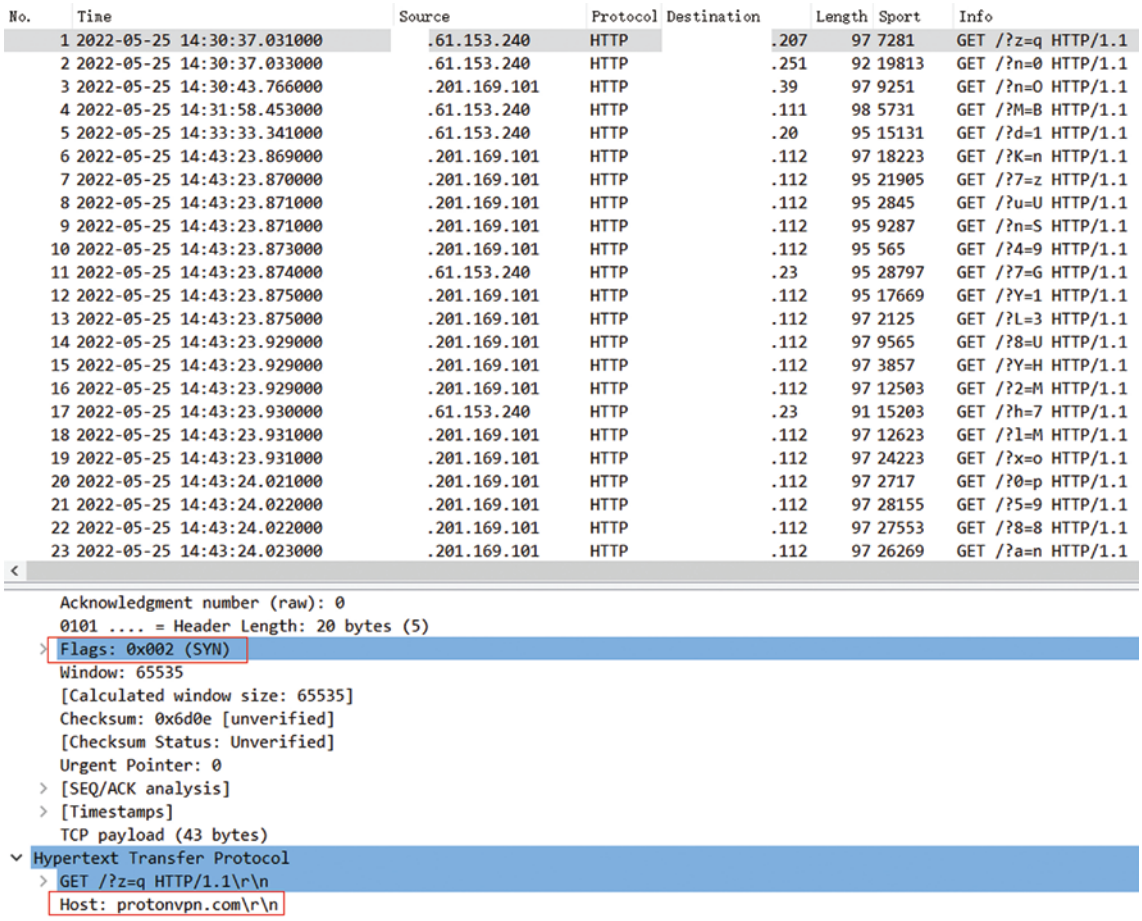
SYN Carpet-Bombing Attacks Triggering HTTP Reflection Attacks Based on Vulnerabilities of Content Security Devices at the Borders of Public Clouds



Two vulnerability exploits were involved during the attacks. The first is the vulnerability exploit related to the session check mechanism. A single SYN packet carrying a payload passes the Layer 4 session check of the content security device and enters the Layer 7 content check process. The second is the vulnerability exploit related to the page in response to non-compliant content — The response page is manually configured, and if it contains a lot of content, the size of related packets will be large. When a content security device detects that a domain name carried in a SYN packet is non-compliant, it replaces the cloud host to return a page to the source IP address of the SYN packet, indicating that the request content is non-compliant. In this way, HTTP reflection attacks are formed. A SYN packet that is less than 100 bytes triggers a response packet that exceeds 700 bytes, forming an over 7-fold amplification effect.

Illegal domain names carried in SYN packets that trigger HTTP reflection attacks include startpage.com, protonvpn.com, expressvpn.com, stake.com, fmovies.to and youporn.com.

Obtained Packets of SYN Carpet-Bombing Attacks that Exploit Content Security Devices to Generate HTTP Reflection Packets



No.	Time	Source	Protocol	Destination	Length	Sport	Info
1	2022-05-25 14:30:37.031000	.61.153.240	HTTP		.207	97 7281	GET /?z=q HTTP/1.1
2	2022-05-25 14:30:37.033000	.61.153.240	HTTP		.251	92 19813	GET /?n=0 HTTP/1.1
3	2022-05-25 14:30:43.766000	.201.169.101	HTTP		.39	97 9251	GET /?n=O HTTP/1.1
4	2022-05-25 14:31:58.453000	.61.153.240	HTTP		.111	98 5731	GET /?M=B HTTP/1.1
5	2022-05-25 14:33:33.341000	.61.153.240	HTTP		.20	95 15131	GET /?d=1 HTTP/1.1
6	2022-05-25 14:43:23.869000	.201.169.101	HTTP		.112	97 18223	GET /?K=n HTTP/1.1
7	2022-05-25 14:43:23.870000	.201.169.101	HTTP		.112	95 21905	GET /?7=z HTTP/1.1
8	2022-05-25 14:43:23.871000	.201.169.101	HTTP		.112	95 2845	GET /?u=U HTTP/1.1
9	2022-05-25 14:43:23.871000	.201.169.101	HTTP		.112	95 9287	GET /?n=S HTTP/1.1
10	2022-05-25 14:43:23.873000	.201.169.101	HTTP		.112	95 565	GET /?4=9 HTTP/1.1
11	2022-05-25 14:43:23.874000	.61.153.240	HTTP		.23	95 28797	GET /?7=G HTTP/1.1
12	2022-05-25 14:43:23.875000	.201.169.101	HTTP		.112	95 17669	GET /?Y=1 HTTP/1.1
13	2022-05-25 14:43:23.875000	.201.169.101	HTTP		.112	97 2125	GET /?L=3 HTTP/1.1
14	2022-05-25 14:43:23.929000	.201.169.101	HTTP		.112	97 9565	GET /?8=U HTTP/1.1
15	2022-05-25 14:43:23.929000	.201.169.101	HTTP		.112	97 3857	GET /?Y=H HTTP/1.1
16	2022-05-25 14:43:23.929000	.201.169.101	HTTP		.112	97 12503	GET /?2=M HTTP/1.1
17	2022-05-25 14:43:23.930000	.61.153.240	HTTP		.23	91 15203	GET /?h=7 HTTP/1.1
18	2022-05-25 14:43:23.931000	.201.169.101	HTTP		.112	97 12623	GET /?l=M HTTP/1.1
19	2022-05-25 14:43:23.931000	.201.169.101	HTTP		.112	97 24223	GET /?x=o HTTP/1.1
20	2022-05-25 14:43:24.021000	.201.169.101	HTTP		.112	97 2717	GET /?0=p HTTP/1.1
21	2022-05-25 14:43:24.022000	.201.169.101	HTTP		.112	97 28155	GET /?5=9 HTTP/1.1
22	2022-05-25 14:43:24.022000	.201.169.101	HTTP		.112	97 27553	GET /?8=8 HTTP/1.1
23	2022-05-25 14:43:24.023000	.201.169.101	HTTP		.112	97 26269	GET /?a=n HTTP/1.1

```
Acknowledgment number (raw): 0
0101 .... = Header Length: 20 bytes (5)
Flags: 0x002 (SYN)
Window: 65535
[Calculated window size: 65535]
Checksum: 0x6d0e [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[SEQ/ACK analysis]
[Timestamps]
TCP payload (43 bytes)
Hypertext Transfer Protocol
GET /?z=q HTTP/1.1\r\n
Host: protonvpn.com\r\n
```

The content security device sends an RST packet to the destination IP address of the SYN packet, that is, the cloud host, in an attempt to release session resources of the cloud host. As a result, the cloud host suffers both SYN carpet-bombing attacks from the attacking device and RST carpet-bombing attacks from the content security device.

Obtained Packets of Reflection Attacks that Exploit Content Security Devices

A content security device replaces a cloud host to return a FIN/PSH/ACK packet to the source IP address of a SYN packet, indicating that the HTTP request content is non-compliant.

The content security device uses the source IP address of a SYN packet to send an RST packet to the cloud host, in an attempt to release session resources of the cloud host.

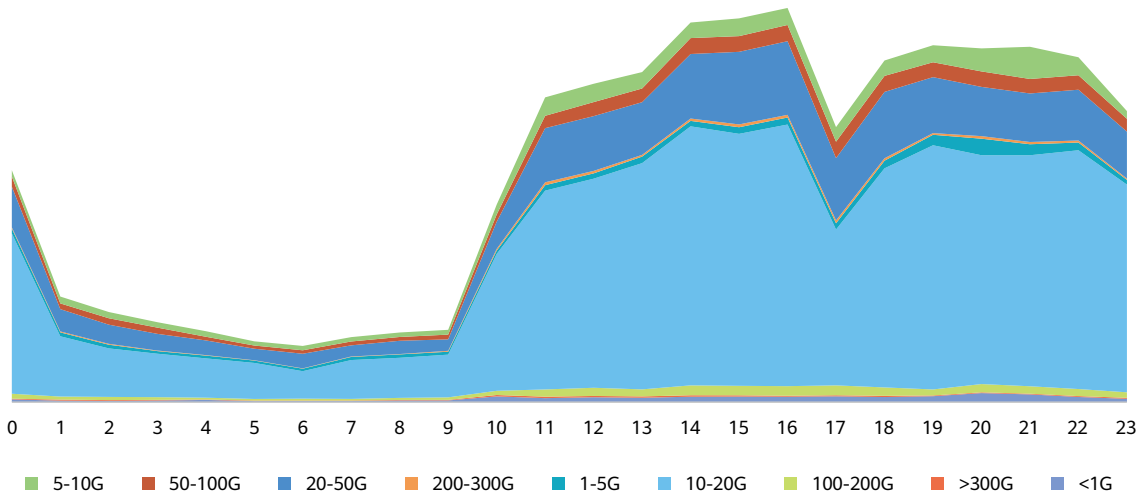
No.	Time	Source	Protocol	Destination	Length	Sport	Info
1	2022-05-25 10:46:18.702174	.196.192.37	TCP	.97	707	80	80 → 23485 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
2	2022-05-25 10:46:18.702272	.41.106.158	TCP	.169	64	12463	12463 → 80 [RST] Seq=1 Win=0 Len=0
3	2022-05-25 10:46:18.702309	.33.96.97	TCP	.86	64	26831	26831 → 80 [RST] Seq=1 Win=0 Len=0
4	2022-05-25 10:46:18.702645	.33.96.97	TCP	.229	64	22931	22931 → 80 [RST] Seq=1 Win=0 Len=0
5	2022-05-25 10:46:18.702703	.33.96.97	TCP	.189	64	2665	2665 → 80 [RST] Seq=1 Win=0 Len=0
6	2022-05-25 10:46:18.702738	.33.96.97	TCP	.78	64	22897	22897 → 80 [RST] Seq=1 Win=0 Len=0
7	2022-05-25 10:46:18.702771	.210.128.141	TCP	.75	707	80	80 → 18125 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
8	2022-05-25 10:46:18.702804	.196.193.7	TCP	.158	707	80	80 → 28609 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
9	2022-05-25 10:46:18.702837	.41.106.158	TCP	.7	64	28609	28609 → 80 [RST] Seq=4294967253 Win=0 Len=0
10	2022-05-25 10:46:18.702890	.196.193.102	TCP	.97	707	80	80 → 27537 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
11	2022-05-25 10:46:18.702929	.41.106.158	TCP	.7	64	28609	28609 → 80 [RST] Seq=1 Win=0 Len=0
12	2022-05-25 10:46:18.702961	.196.194.140	TCP	.97	707	80	80 → 29395 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
13	2022-05-25 10:46:18.702994	.196.195.209	TCP	.75	707	80	80 → 29109 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
14	2022-05-25 10:46:18.703025	.33.2.75	TCP	.209	64	29109	29109 → 80 [RST] Seq=4294967259 Win=0 Len=0
15	2022-05-25 10:46:18.703056	.196.195.124	TCP	.97	707	80	80 → 4873 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
16	2022-05-25 10:46:18.703104	.33.96.97	TCP	.228	64	7695	7695 → 80 [RST] Seq=1 Win=0 Len=0
17	2022-05-25 10:46:18.703140	.196.196.224	TCP	.97	707	80	80 → 3653 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
18	2022-05-25 10:46:18.703173	.196.194.187	TCP	.75	707	80	80 → 1933 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
19	2022-05-25 10:46:18.703237	.33.96.97	TCP	.224	64	3653	3653 → 80 [RST] Seq=4294967253 Win=0 Len=0
20	2022-05-25 10:46:18.703760	.196.199.12	TCP	.75	707	80	80 → 18905 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
21	2022-05-25 10:46:18.703821	.196.193.191	TCP	.75	707	80	80 → 23585 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
22	2022-05-25 10:46:18.703856	.196.194.14	TCP	.97	707	80	80 → 14351 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
23	2022-05-25 10:46:18.703890	.33.96.97	TCP	.14	64	14351	14351 → 80 [RST] Seq=1 Win=0 Len=0
24	2022-05-25 10:46:18.703921	.33.2.75	TCP	.76	64	17269	17269 → 80 [RST] Seq=1 Win=0 Len=0
25	2022-05-25 10:46:18.703952	.196.195.32	TCP	.75	707	80	80 → 29937 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
26	2022-05-25 10:46:18.704022	.41.106.158	TCP	.182	64	4505	4505 → 80 [RST] Seq=1 Win=0 Len=0
27	2022-05-25 10:46:18.704065	.196.192.191	TCP	.97	707	80	80 → 11045 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
28	2022-05-25 10:46:18.704099	.33.96.97	TCP	.79	64	24509	24509 → 80 [RST] Seq=1 Win=0 Len=0
29	2022-05-25 10:46:18.704131	.196.192.211	TCP	.97	707	80	80 → 12327 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
30	2022-05-25 10:46:18.704162	.33.96.97	TCP	.78	64	1559	1559 → 80 [RST] Seq=1 Win=0 Len=0
31	2022-05-25 10:46:18.704194	.196.193.103	TCP	.75	707	80	80 → 13513 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
32	2022-05-25 10:46:18.704248	.196.195.255	TCP	.75	707	80	80 → 28665 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]
33	2022-05-25 10:46:18.704290	.196.195.77	TCP	.97	707	80	80 → 18081 [FIN, PSH, ACK] Seq=1 Ack=1 Win=8192 Len=649[Packet size limited during capture]

When attacks occur, multiple class C IP addresses on the public cloud suffer SYN carpet-bombing attacks. High-rate SYN packets consume session resources of the cloud platform and occupy the inbound bandwidth of the cloud host. HTTP reflection traffic occupies the outbound bandwidth of the public cloud, and RST carpet-bombing traffic occupies the inbound bandwidth of the cloud host.

3.1.5 Attack Occurrence Period

The time period that most attacks occur in 2022 is the same as that in previous years — consistent with the active time of most netizens. This is to maximize the attack effect at the lowest attack cost.

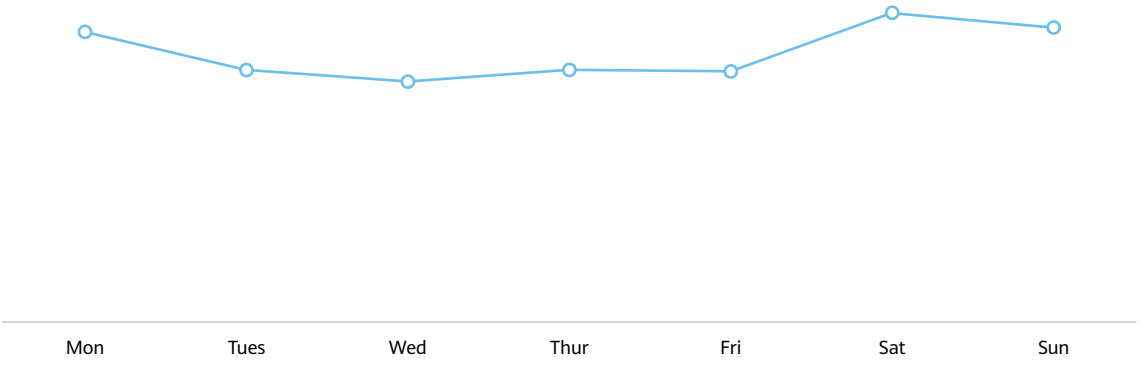
2022 Attack Occurrence Periods



Source: China Telecom Cybersecurity

According to the weekly attack distribution statistics, attacks are evenly distributed within a week, the number of attacks occurred from Tuesday to Friday slightly decreases, and attacks are active at weekends.

2022 Number of Attacks Occurred in a Week

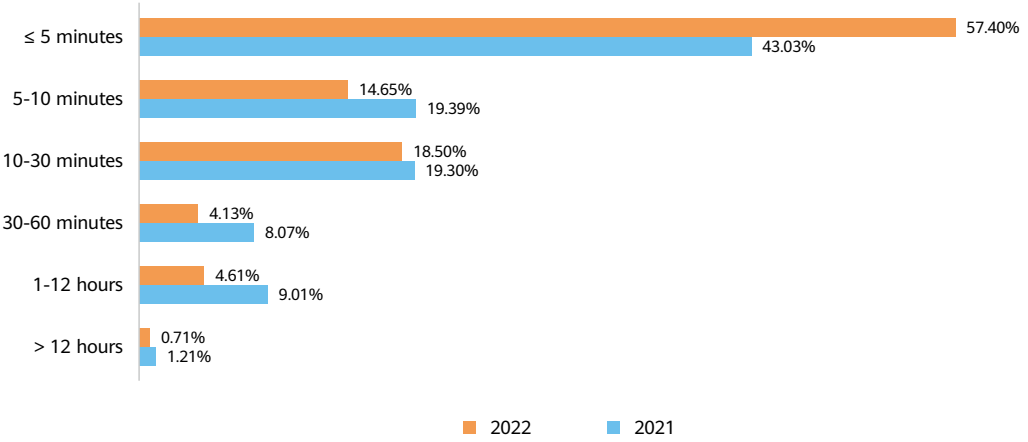


Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

3.1.6 Attack Duration

According to the distribution of network-layer attack duration, most attacks tend to be initiated in a "fast flooding" manner. In 2021, the proportion of attacks lasted for ≤ 5 minutes was 43.03%. In 2022, the proportion increased to 57.40%. Such "fast flooding" attacks challenge the automation rate of the defense system and the response speed of the security team.

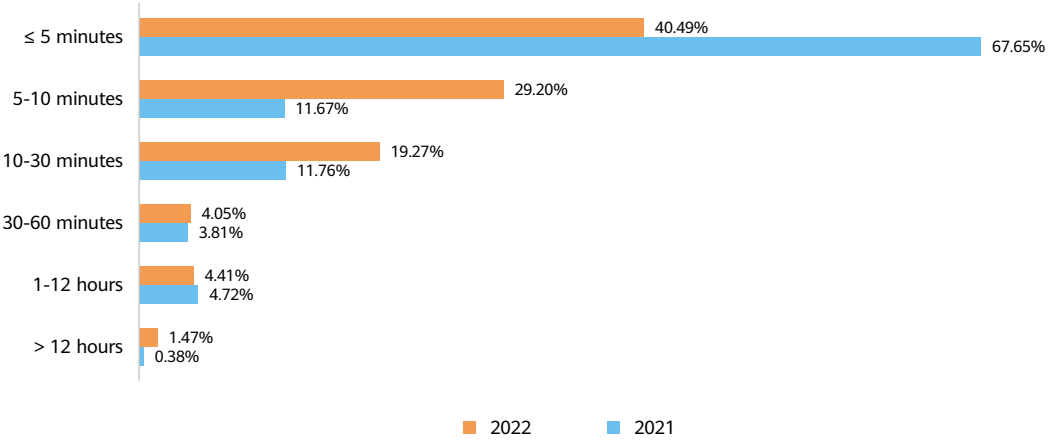
2021-2022 Distribution of Network-Layer Attack Duration



Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

Application-layer attacks also display an obvious trend of "fast flooding". In 2022, application-layer attacks lasted for ≤ 10 minutes accounted for 69.69%. Botnets are less likely to be detected when attacks last for a shorter period of time. Moreover, short-duration attacks challenge the automation rate of the defense system and the response speed of the security team.

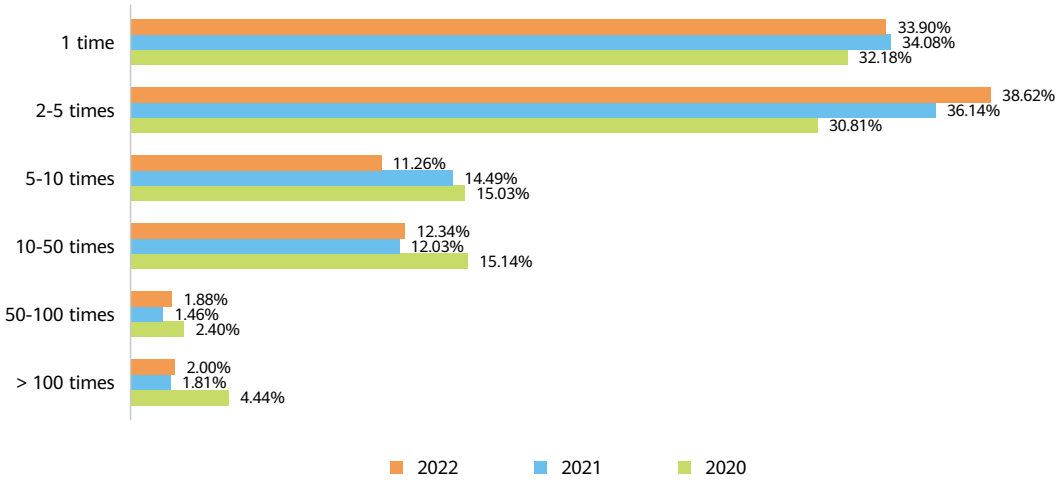
2021-2022 Distribution of Network-Layer Attack Duration



3.1.7 Attack Persistence

The proportion of IP addresses that have been attacked for multiple times exceeds 65% in the past three years. This indicates that most IP addresses may be attacked repeatedly once they become attack targets.

2020-2022 Number of Times that Attacks Recur



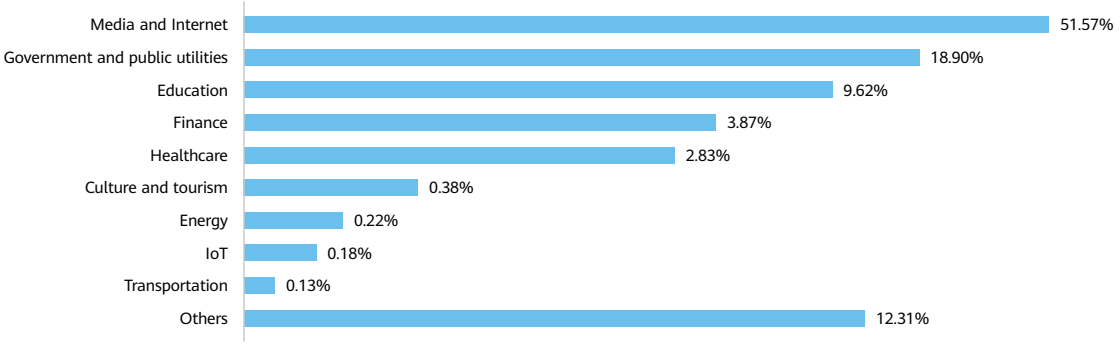
Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

3.1.8 Attack Target Distribution by Industry

According to China Unicom Digital Tech's statistics on the number of attacks suffered by customers in various industries, the media and Internet industries were still hit the hardest by DDoS attacks in 2022. The number of attacks on the two industries accounted for 51.57% of total attacks. The media and Internet industry includes sectors such as gaming, e-commerce, Internet finance, and social networking. Due to fierce competition within industries, the media and Internet industries have always been hit the hardest by DDoS attacks.

In addition, government and public utilities, education, finance, and healthcare industries are also key targets of DDoS attacks. The number of attacks on these industries accounted for 18.9%, 9.62%, 3.87%, and 2.83% of total attacks, respectively.

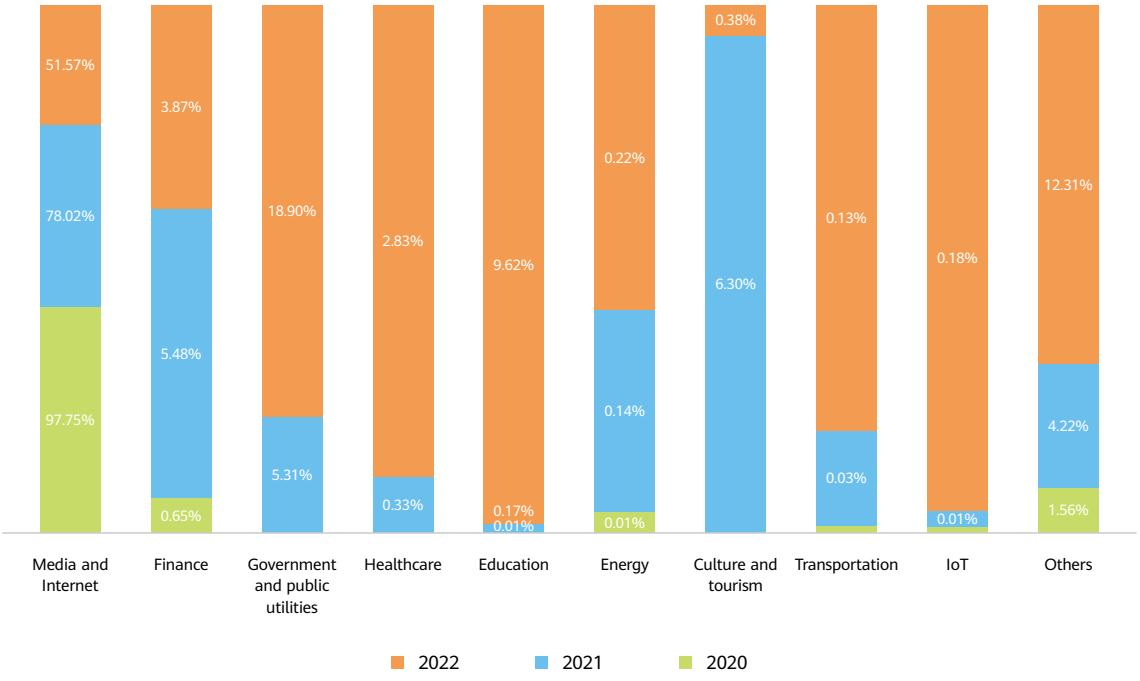
2022 Attack Target Distribution by Industry



Source: China Unicom Digital Tech

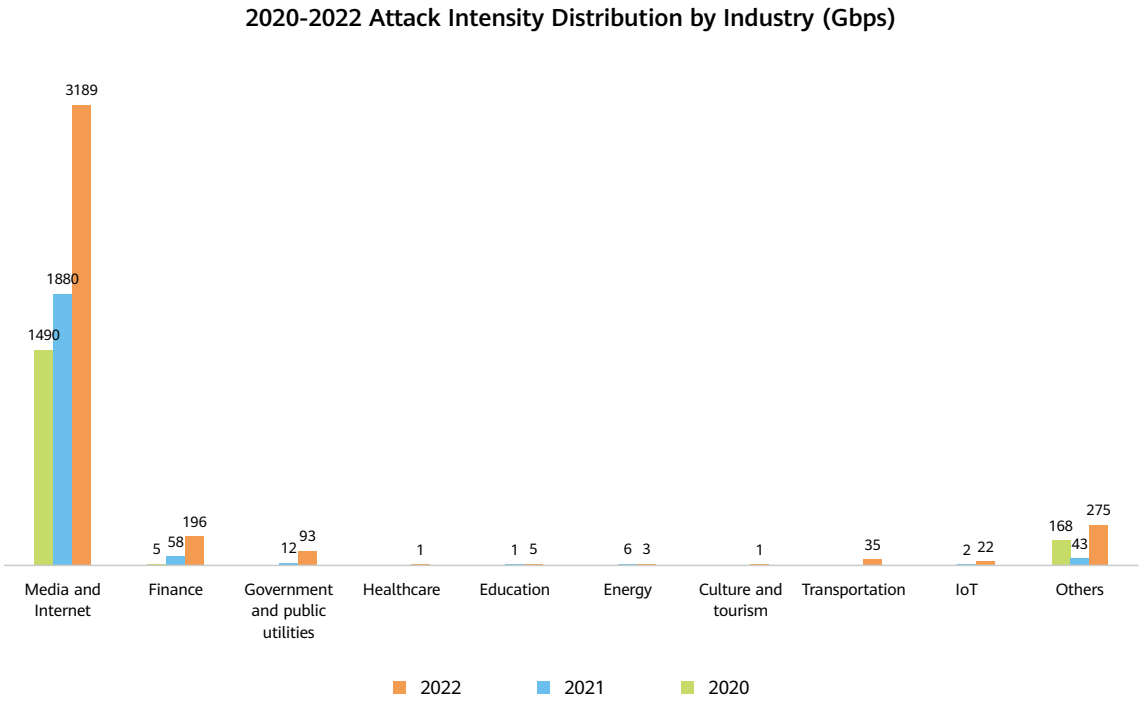
According to the statistics on attack target distribution by industry from 2020 to 2022, as services of various industries are migrated to the cloud, the Internet-exposed attack surface is expanding, and the number of DDoS attacks is on the rise year by year. Compared with the statistics in 2021, in 2022, the proportion of attacks targeting the education industry, healthcare industry, as well as the government and public utilities industry increased by 56.6 times, 8.6 times, and 3.6 times, respectively. In addition, the Industrial Internet, as an emerging field, becomes a new target of DDoS attacks. In 2022, attacks targeting the Industrial Internet was 18 times that of 2021.

2020-2022 Attack Target Distribution by Industry



Source: China Unicom Digital Tech

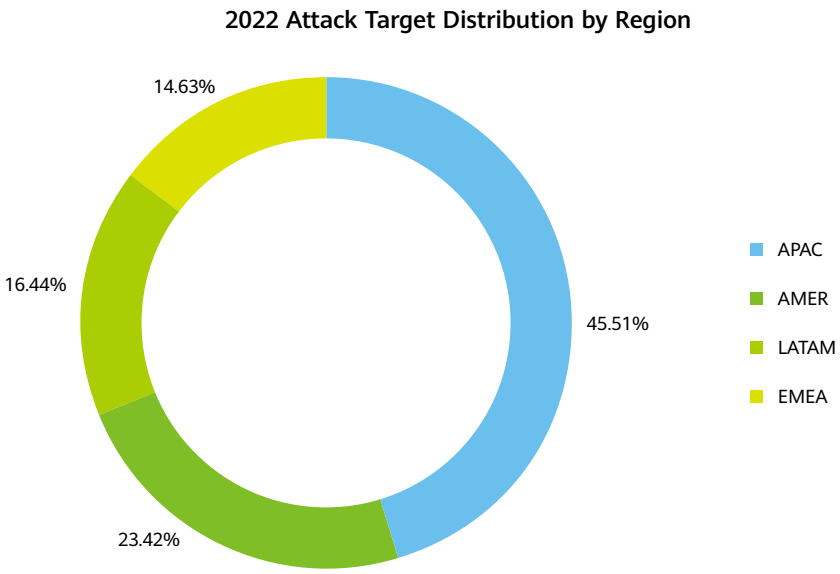
According to the statistics on attack intensity distribution by industry from 2020 to 2022, the intensity of attacks on almost all industries increases year by year. Due to fierce competition in the industry, the media and Internet industry is the most vulnerable to ultra-large scale DDoS attacks. In the past three years, the industry has suffered the most violent attacks among all industries. In 2022, attacks targeting this industry even peaked at 3.189 Tbps. The finance industry has always suffered massive attacks, and the intensity of attacks has increased year by year. In 2022, attacks targeting this industry peaked at 196 Gbps. In 2022, the government and public utilities, transportation, and other industries also suffered high-intensity attacks. Attacks targeting the government and public utilities industry peaked at 93 Gbps, and for the transportation industry and the Industrial Internet, the peak bandwidth was 35 Gbps and 22 Gbps respectively.



Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

3.1.9 Attack Target Distribution by Region

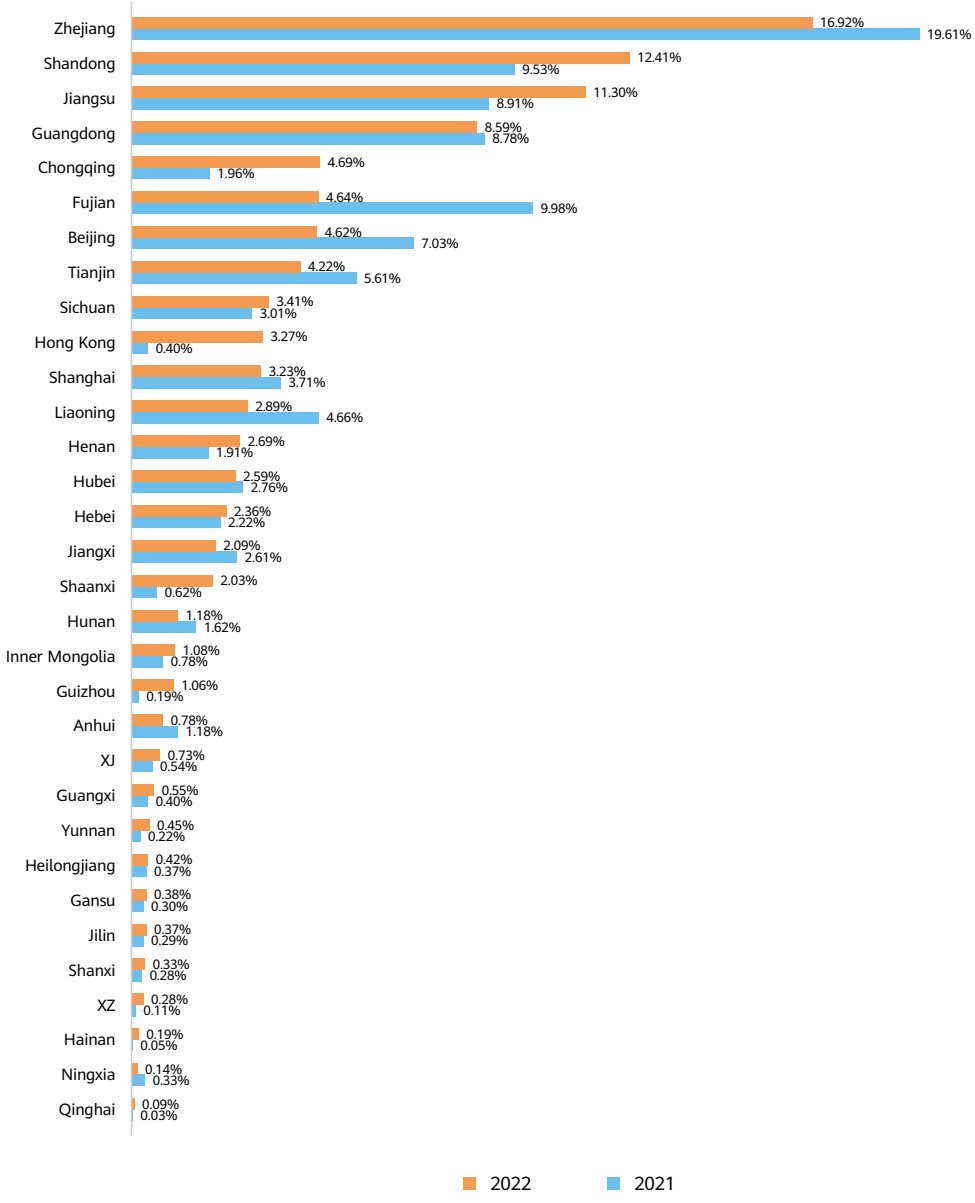
According to the attack target distribution by region in 2022, the number of attacks occurred in APAC, AMER, LATAM, and EMEA accounted for 45.51%, 23.42%, 16.44%, and 14.63% of the total, respectively. The number of attacks occurred in APAC and AMER is the largest.



Source: Baidu Security, Nexusguard, Huawei

In 2022, the top 3 attack targets in China were Zhejiang, Shandong, and Jiangsu.

2021-2022 Attack Target Distribution by Region (China)

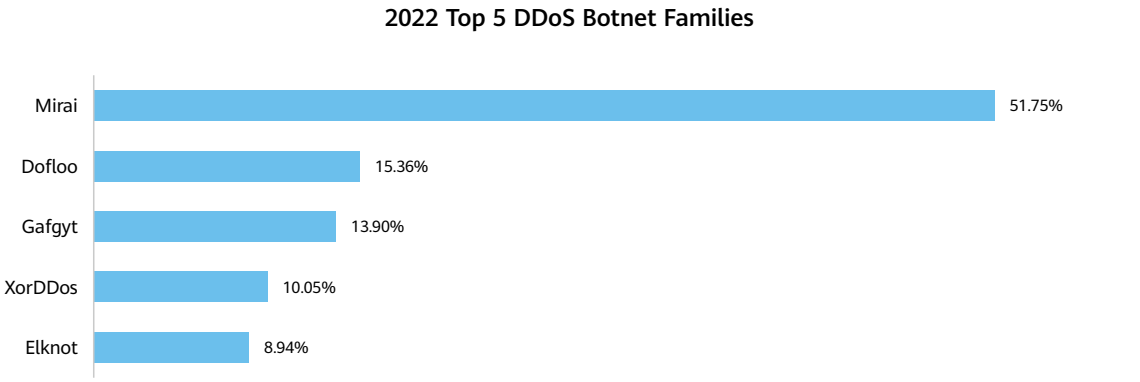


Source: China Telecom Cybersecurity, China Unicom Digital Tech, Huawei

3.2 DDoS Botnet Situation

3.2.1 Botnet Family Distribution

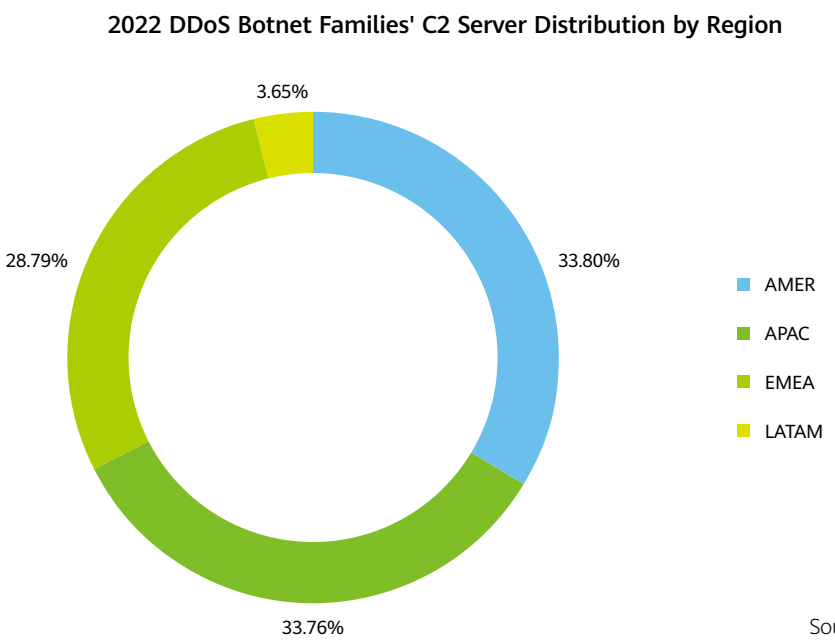
IoT-based botnets and Linux-based botnets are the most active. The top 5 botnet families ranked by the number of active C2 servers are Mirai, Dofloo, Gafgyt, XorDDoS, and Elknot. Mirai and Gafgyt are typical IoT-based botnets, while XorDDoS, Dofloo, and Elknot are typical Linux-based botnets.



Source: Baidu Security

3.2.2 C2 Server Distribution by Region

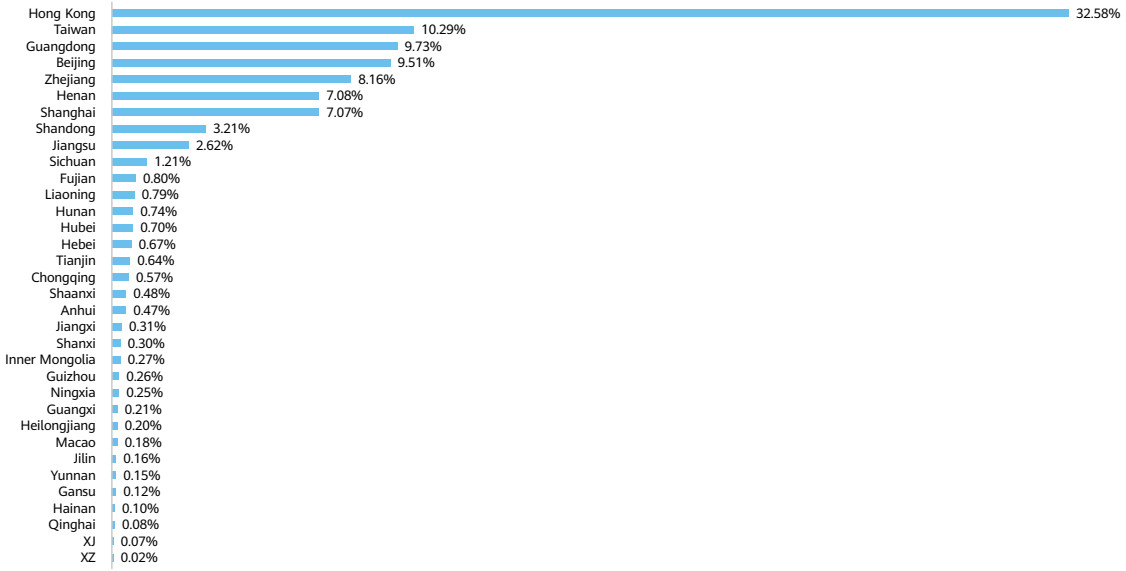
According to the botnet C2 server distribution by region in 2022, C2 servers are evenly distributed in AMER, APAC, and EMEA, except LATAM.



Source: Baidu Security

The top 3 regions with the largest number of botnet C2 servers in China are Hong Kong, Taiwan, and Guangdong.

2022 DDoS Botnet Families' C2 Server Distribution by Region (China)



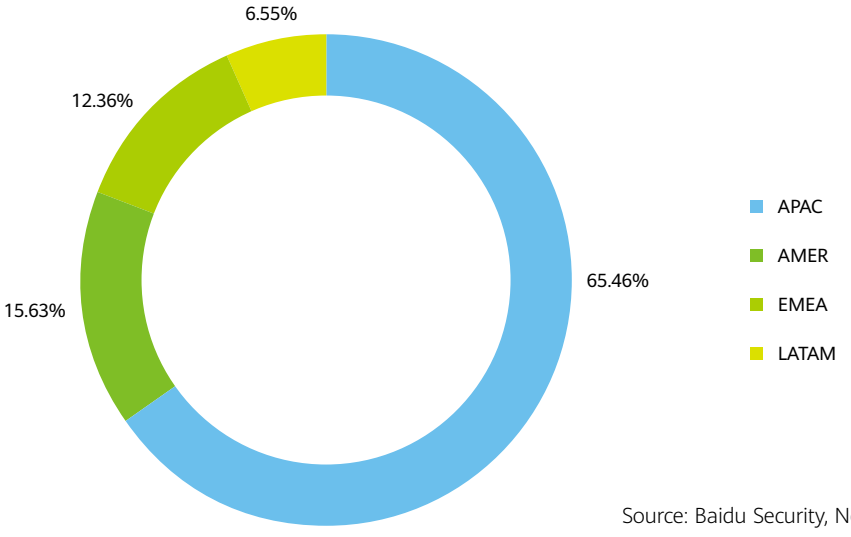
Source: Baidu Security

3.3 DDoS Attack Source Situation

3.3.1 Zombie Distribution by Region

According to the zombie distribution by region, APAC, AMER, and EMEA have the largest number of zombies, accounting for 65.46%, 15.63%, and 12.36% of the total, respectively.

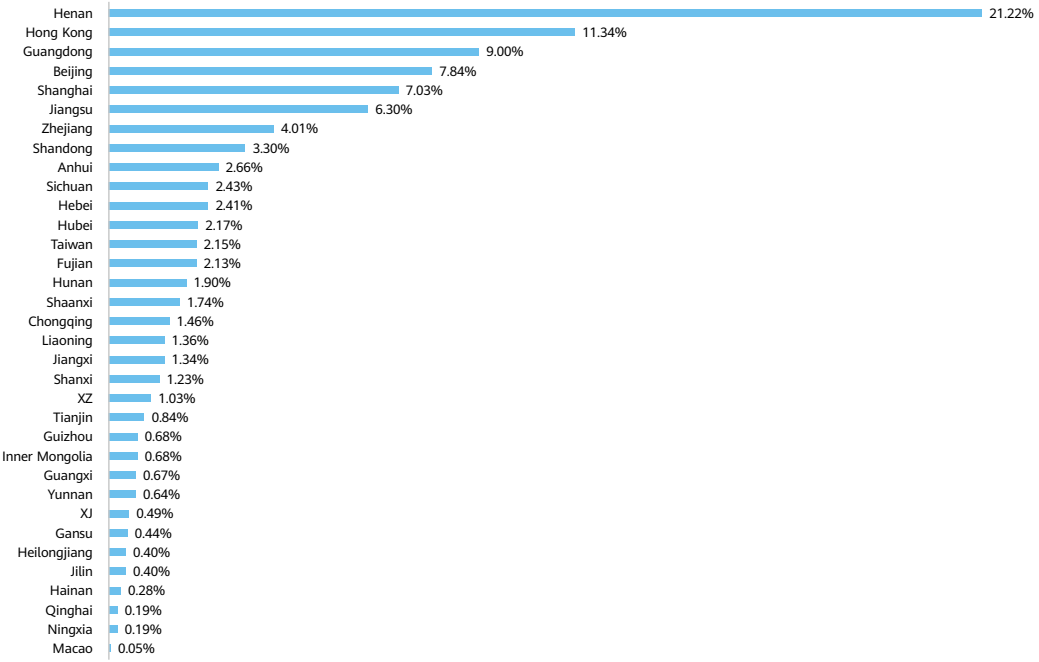
2022 DDoS Zombie Distribution by Region



Source: Baidu Security, Nexusguard, Huawei

The top 3 regions with the largest number of zombies in China are Henan, Hong Kong, and Guangdong.

2022 DDoS Zombie Distribution by Region (China)

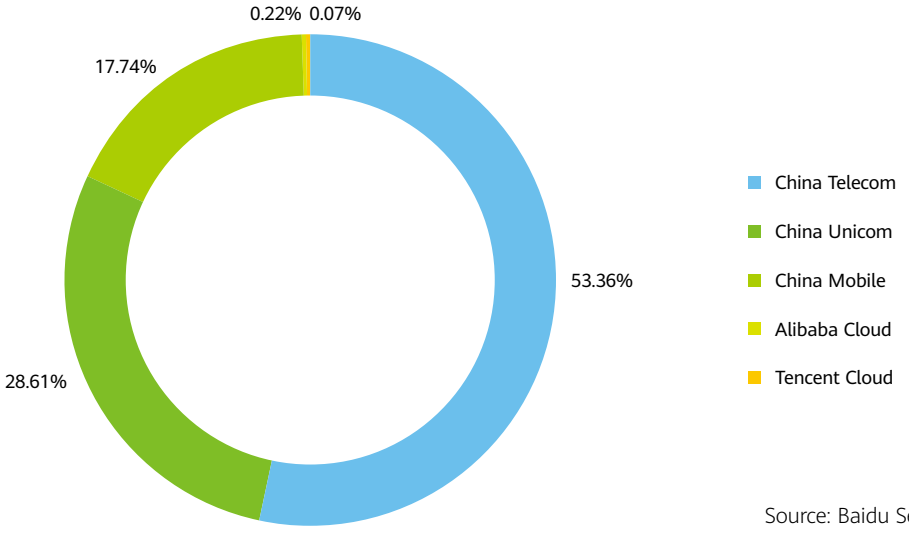


Source: Baidu Security, Huawei

3.3.2 Zombie Distribution by Carrier

In China, China Telecom, China Unicom, and China Mobile, in a descending order, have the largest number of zombies on their networks.

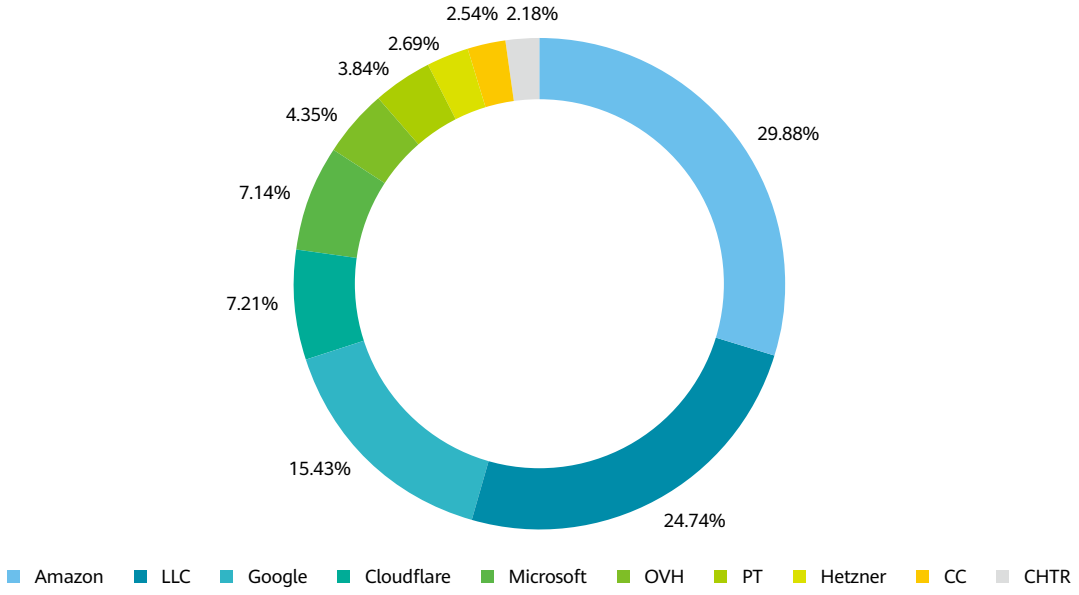
2022 Top 5 Chinese Carriers with the Largest Number of Zombies on Their Networks



Source: Baidu Security, Huawei

Outside China, the victims are Amazon, LLC, and Google. The number of zombies on public clouds accounts for a large proportion, indicating that public cloud providers mainly focus on cloud infrastructure security, and both public cloud providers and tenants do not invest enough in improving cloud host security.

2022 Top 10 Carriers (Outside China) with the Largest Number of Zombies on Their Networks



Source: Baidu Security, Huawei

04 Analysis on Typical DDoS Attacks

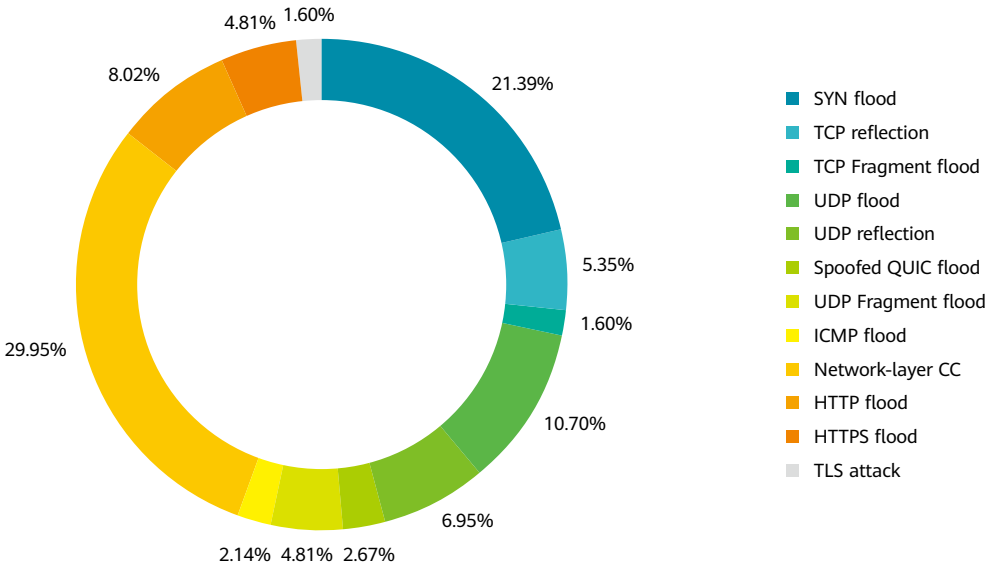
4.1 API DDoS Attack Situation

APIs are regarded as one of the core technologies supporting mobile applications, IoT, and cloud services. Application of APIs has become ubiquitous in our daily life, such as online shopping, takeout ordering, express delivery, hospital registration, like sending during live streams, and remote control of smart home appliances. API calls currently account for more than 83% of all Internet traffic according to Akamai[4]. Cloudflare claims that 55% of traffic traversing its network is related to APIs[5]. Due to the large quantity of APIs and enterprises' inadequate attention to API security, API attacks have become one of the biggest threats faced by enterprises. Cloudflare also claims that its network security devices block more traffic attacking APIs than that attacking websites. This indicates that APIs have become the main target of network attacks[5].

4.1.1 Attack Type Distribution

By tracing the DDoS attacks targeting APIs, we find that APIs suffer most of the attacks that have occurred on the Internet. In 2022, the top 5 types of attacks targeting APIs were network-layer CC, SYN flood, UDP flood, HTTP flood, and UDP reflection attacks.

Distribution of DDoS Attack Types on APIs



4.1.2 Typical Attack Analysis

1. Network-Layer CC Attack

Network-layer CC attack is most frequently used because it is difficult to defend against. High-rate ACK packets consume the performance of the API server and occupy the network bandwidth. To evade the defense system, attackers continuously change the size of ACK packets.

» **Method 1: Using ACK flood with large packets to challenge the defense speed**

Attacking an API Using Network-Layer CC with Large ACK Packets

No.	Time	Source	Protocol	Destination	Length	Sport	Info
1	2022-12-01 22:10:55.061285	49.213.203.50	TCP	.4	74	55911	55911 → 443 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=150235970 TSecr=0 WS=32
2	2022-12-01 22:10:55.066935	.4	TCP	49.213.203.50	70	443	443 → 55911 [SYN, ACK] Seq=0 Ack=1 Win=28200 Len=0 MSS=1394 SACK_PERM=1 WS=512
3	2022-12-01 22:10:55.574392	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
4	2022-12-01 22:10:55.574423	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
5	2022-12-01 22:10:55.574440	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
6	2022-12-01 22:10:55.574441	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
7	2022-12-01 22:10:55.574446	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
8	2022-12-01 22:10:55.574454	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
9	2022-12-01 22:10:55.574478	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
10	2022-12-01 22:10:55.574489	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
11	2022-12-01 22:10:55.574502	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
12	2022-12-01 22:10:55.575435	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
13	2022-12-01 22:10:55.575442	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
14	2022-12-01 22:10:55.575454	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
15	2022-12-01 22:10:55.575479	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
16	2022-12-01 22:10:55.575515	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
17	2022-12-01 22:10:55.575533	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
18	2022-12-01 22:10:55.575574	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
19	2022-12-01 22:10:55.575587	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
20	2022-12-01 22:10:55.575594	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
21	2022-12-01 22:10:55.576462	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
22	2022-12-01 22:10:55.576473	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
23	2022-12-01 22:10:55.576477	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
24	2022-12-01 22:10:55.576504	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
25	2022-12-01 22:10:55.576519	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
26	2022-12-01 22:10:55.576551	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
27	2022-12-01 22:10:55.576558	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
28	2022-12-01 22:10:55.576568	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
29	2022-12-01 22:10:55.576577	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
30	2022-12-01 22:10:55.577498	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
31	2022-12-01 22:10:55.577514	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
32	2022-12-01 22:10:55.577525	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
33	2022-12-01 22:10:55.577529	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
34	2022-12-01 22:10:55.577541	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data
35	2022-12-01 22:10:55.577549	49.213.203.50	SSL	.4	1454	55911	[TCP ACKed unseen segment] [TCP Previous segment not captured] , Continuation Data

After the attacker establishes a TCP connection with the API server through the socket, the attacker uses a raw socket to send high-rate large ACK packets whose payload exceeds 1000 bytes to the server. The attack ramp-up speed is fast, taking only 10 seconds, and the peak bandwidth of attack traffic can surge to over 500 Gbps, challenging the response speed of the defense system.

» **Method 2: Using ACK flood with small packets to evade large packet detection**

To evade the large packet detection mechanism of the defense system, after the attacker establishes a TCP connection with the API server through the socket, the attacker uses the raw socket to send small ACK packets without payload to the server at a high speed, consuming server performance.

Attacking an API Using Network-Layer CC with Small ACK Packets

No.	Time	Source	Destination	srcport	length	info
37	2022-04-20 16:27:20.591000	221.15.7.149	103.	59301	54	59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
39	2022-04-20 16:27:20.592000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#1] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
40	2022-04-20 16:27:20.593000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#2] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
65	2022-04-20 16:27:20.601000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#3] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
66	2022-04-20 16:27:20.602000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#4] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
67	2022-04-20 16:27:20.602000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#5] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
68	2022-04-20 16:27:20.602000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#6] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
69	2022-04-20 16:27:20.603000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#7] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
70	2022-04-20 16:27:20.603000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#8] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
71	2022-04-20 16:27:20.603000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#9] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
72	2022-04-20 16:27:20.604000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#10] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
73	2022-04-20 16:27:20.604000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#11] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
74	2022-04-20 16:27:20.605000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#12] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
75	2022-04-20 16:27:20.605000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#13] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
76	2022-04-20 16:27:20.605000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#14] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
77	2022-04-20 16:27:20.606000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#15] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
78	2022-04-20 16:27:20.606000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#16] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
79	2022-04-20 16:27:20.606000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#17] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
108	2022-04-20 16:27:20.617000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#18] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
110	2022-04-20 16:27:20.618000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#19] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
111	2022-04-20 16:27:20.618000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#20] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
113	2022-04-20 16:27:20.619000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#21] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
114	2022-04-20 16:27:20.619000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#22] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
115	2022-04-20 16:27:20.619000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#23] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
116	2022-04-20 16:27:20.620000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#24] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
249	2022-04-20 16:27:20.668000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#25] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
252	2022-04-20 16:27:20.669000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#26] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
253	2022-04-20 16:27:20.670000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#27] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
254	2022-04-20 16:27:20.670000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#28] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
255	2022-04-20 16:27:20.670000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#29] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
256	2022-04-20 16:27:20.671000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#30] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0
257	2022-04-20 16:27:20.671000	221.15.7.149	103.	59301	54	[TCP Dup ACK 37#31] 59301→443 [ACK] Seq=1 Ack=1 Win=513 Len=0

» **Method 3: Using ACK flood with packets of randomly changing sizes, and simulating the interaction between the client and server to evade defense**

Disguised as the client, the network-layer CC attacks targeting APIs often uses ACK packets with randomly changed payload content and length to interact with the server and confuse the defense system. In this way, the server performance is exhausted and the network bandwidth is consumed.

Attacking an API Using Network-Layer CC with Variable-Size ACK Packets

Time	Source	Protocol	Destination	Length	Sport	Info
2022-05-27 03:37:27.062000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.062000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.063000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=63537 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.063000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=49097 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.064000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=50541 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.064000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=70757 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.064000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=53429 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.065000	115.171.126.239	TCP	.142	1328	64274	[TCP Retransmission] 64274 → 443 [ACK] Seq=51985 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.065000	115.171.126.239	TCP	.142	1328	64274	[TCP Retransmission] 64274 → 443 [ACK] Seq=46209 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.066000	115.171.126.239	TCP	.142	1328	64274	[TCP Retransmission] 64274 → 443 [ACK] Seq=57761 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.066000	115.171.126.239	TCP	.142	1328	64320	64320 → 443 [ACK] Seq=1 Ack=1 Win=4096 Len=1274 [TCP segment of a reassembled PDU]
2022-05-27 03:37:27.066000	115.171.126.239	TLSv1.2	.142	830	64320	[TCP Previous segment not captured] , Application Data
2022-05-27 03:37:27.067000	115.171.126.239	TCP	.142	997	64320	[TCP Out-Of-Order] 64320 → 443 [PSH, ACK] Seq=1445 Ack=1 Win=4096 Len=943 [TCP segment of a reassembled PDU]
2022-05-27 03:37:27.067000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.068000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.068000	115.171.126.239	TCP	.142	54	64328	64328 → 443 [ACK] Seq=1 Ack=1 Win=4084 Len=0
2022-05-27 03:37:27.068000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.069000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.069000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.070000	115.171.126.239	TCP	.142	54	64328	[TCP Previous segment not captured] 64328 → 443 [FIN, ACK] Seq=32 Ack=1 Win=4096 Len=0
2022-05-27 03:37:27.070000	115.171.126.239	TCP	.142	85	64328	[TCP Out-Of-Order] 64328 → 443 [PSH, ACK] Seq=1 Ack=1 Win=4096 Len=31
2022-05-27 03:37:27.070000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.071000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.071000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.072000	115.171.126.239	TLSv1.2	.142	879	64314	Application Data
2022-05-27 03:37:27.072000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data
2022-05-27 03:37:27.072000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=105413 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.073000	115.171.126.239	TCP	.142	1328	64274	[TCP Out-Of-Order] 64274 → 443 [ACK] Seq=101081 Ack=1 Win=4096 Len=1274
2022-05-27 03:37:27.073000	115.171.126.239	SSL	.142	1328	64274	[TCP Previous segment not captured] , Continuation Data

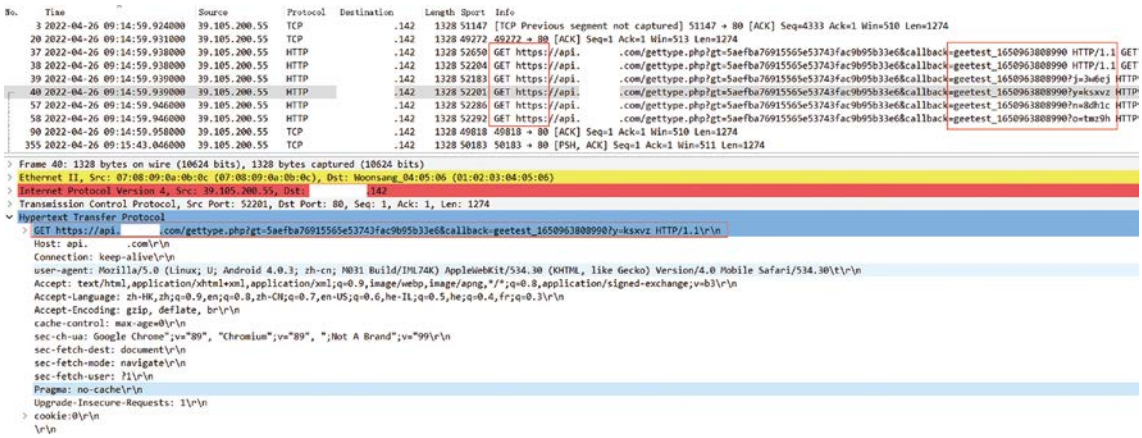
To increase defense costs, attackers usually use both high-rate and low-rate network-layer CC attacks to attack APIs. High-rate attacks challenge the response speed of the defense system, and low-rate attacks increase the defense difficulty.

2. HTTP Application-Layer Attacks

Application-layer attacks on APIs can be classified into attacks on API servers and attacks on specific APIs.

» **Using illegitimate requests to attack the API server**

Attacking an API server: Sending HTTPS Requests to Port 80



```
No.   Time                       Source          Protocol  Destination  Length Sport  Info
    3 2022-04-26 09:14:59.924000 39.105.200.55   TCP                .142   1328 51147  [TCP Previous segment not captured] 51147 → 80 [ACK] Seq=4333 Ack=1 Win=510 Len=1274
   20 2022-04-26 09:14:59.931000 39.105.200.55   TCP                .142   1328 49272  49272 → 80 [ACK] Seq=1 Ack=1 Win=513 Len=1274
   37 2022-04-26 09:14:59.938000 39.105.200.55   HTTP               .142   1328 52650  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990 HTTP/1.1 GET
   38 2022-04-26 09:14:59.938000 39.105.200.55   HTTP               .142   1328 52204  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990 HTTP/1.1 GET
   39 2022-04-26 09:14:59.939000 39.105.200.55   HTTP               .142   1328 52183  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990?j=3w6ej HTTP
   40 2022-04-26 09:14:59.939000 39.105.200.55   HTTP               .142   1328 52201  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990?y=ksxvz HTTP
   57 2022-04-26 09:14:59.946000 39.105.200.55   HTTP               .142   1328 52286  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990?n=8dh1c HTTP
   58 2022-04-26 09:14:59.946000 39.105.200.55   HTTP               .142   1328 52292  GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990?o=tmz9h HTTP
   90 2022-04-26 09:14:59.958000 39.105.200.55   TCP                .142   1328 49818  49818 → 80 [ACK] Seq=1 Ack=1 Win=510 Len=1274
  355 2022-04-26 09:15:43.046000 39.105.200.55   TCP                .142   1328 50183  50183 → 80 [PSH, ACK] Seq=1 Ack=1 Win=511 Len=1274
> Frame 40: 1328 bytes on wire (10624 bits), 1328 bytes captured (10624 bits)
> Ethernet II, Src: 07:08:09:0a:0b:0c (07:08:09:0a:0b:0c), Dst: Woonsang_04:05:06 (01:02:03:04:05:06)
> Internet Protocol Version 4, Src: 39.105.200.55, Dst:          .142
> Transmission Control Protocol, Src Port: 52201, Dst Port: 80, Seq: 1, Ack: 1, Len: 1274
v Hypertext Transfer Protocol
  > GET https://api.       .com/gettype.php?gt=5aefba76915565e53743fac9b95b33e6&callback=geetest_1650963808990?y=ksxvz HTTP/1.1\r\n
    Host: api.       .com\r\n
    Connection: keep-alive\r\n
    user-agent: Mozilla/5.0 (Linux; U; Android 4.0.3; zh-cn; M031 Build/IML74K) AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari/534.30\t\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3\r\n
    Accept-Language: zh-HK,zh;q=0.9,en;q=0.8,zh-CN;q=0.7,en-US;q=0.6,he-IL;q=0.5,he;q=0.4,fr;q=0.3\r\n
    Accept-Encoding: gzip, deflate, br\r\n
    cache-control: max-age=0\r\n
    sec-ch-ua: Google Chrome";v="89", "Chromium";v="89", ";Not A Brand";v="99\r\n
    sec-fetch-dest: document\r\n
    sec-fetch-mode: navigate\r\n
    sec-fetch-user: ?1\r\n
    Pragma: no-cache\r\n
    Upgrade-Insecure-Requests: 1\r\n
  > cookie:0\r\n
    \r\n
```

As shown in the preceding figure, the attacker attempts to send HTTPS requests in clear text to port 80 for attacking the API server. If the fault tolerance of the API server during its setup process is low, the attack may cause abnormal server processing.

Attacking an API Server:
Requesting the Root Directory Using Variable Methods and Randomly Changing Parameters

```
Time                       Source          Protocol  Destination  Length Sport  Info
2022-02-08 13:17:06.356000 218.244.147.59  HTTP               .142    596 36641  HEAD /?magBXfv0cEMPdOul=Tssee7aX&w00d6=DVh&bJ6vWhBzdM2qnRaLhOL=DVgiIY0 HTTP/1.1
2022-02-08 13:17:06.856000 218.244.147.59  HTTP               .142    671 36676  PUT /?oeFqZM01uOAAwnyX=bGLGClYa&vaaQw=JDg&zxZDcFNtpwxjnhg8E3t=sBIrwOm HTTP/1.1
2022-02-08 13:17:07.188000 218.244.147.59  HTTP               .142    671 36720  PUT /?oeFqZM01uOAAwnyX=bGLGClYa&vaaQw=JDg&zxZDcFNtpwxjnhg8E3t=sBIrwOm HTTP/1.1
2022-02-08 13:17:09.234000 218.244.147.59  HTTP               .142    671 36722  PUT /?oeFqZM01uOAAwnyX=bGLGClYa&vaaQw=JDg&zxZDcFNtpwxjnhg8E3t=sBIrwOm HTTP/1.1
2022-02-08 13:17:09.877000 218.244.147.59  HTTP               .142    633 36611  POST /?EyLS0HTVNRk05CRS=M6jaVSre&gd1Jx=L0X&NvruwHL7h5YkspYOXt4=k8GWciV HTTP/1.1
2022-02-08 13:17:09.997000 218.244.147.59  HTTP               .142    651 36608  DELETE /?Vgu2TWjh3PAAxol3=56Oof1xe&7W3FY=V7I&9PeraX5Q1p3jbZSsJax=cPvLVDD HTTP/1.1
2022-02-08 13:17:10.444000 218.244.147.59  HTTP               .142    651 36743  DELETE /?Vgu2TWjh3PAAxol3=56Oof1xe&7W3FY=V7I&9PeraX5Q1p3jbZSsJax=cPvLVDD HTTP/1.1
2022-02-08 13:17:10.898000 218.244.147.59  HTTP               .142    651 36745  DELETE /?Vgu2TWjh3PAAxol3=56Oof1xe&7W3FY=V7I&9PeraX5Q1p3jbZSsJax=cPvLVDD HTTP/1.1
2022-02-08 13:17:45.577000 218.244.147.59  HTTP               .142    671 36728  PUT /?oeFqZM01uOAAwnyX=bGLGClYa&vaaQw=JDg&zxZDcFNtpwxjnhg8E3t=sBIrwOm HTTP/1.1
2022-02-08 13:17:47.274000 218.244.147.59  HTTP               .142    631 35930  HEAD /?YgEcMZ2hzP9iob2K=sSkpN5wZ&BnEvn=dcM&rfQXbUYrx9xc6ZleQDP=Tl6AVwp HTTP/1.1
2022-02-08 13:17:47.387000 218.244.147.59  HTTP               .142    632 36534  HEAD /?ccbh106N0czTh9IZ=aOUVgRhI&olTcI=UxJ&qVtAUcTmaMwrMxbTUga=N8niR3k HTTP/1.1
2022-02-08 13:17:48.603000 218.244.147.59  HTTP               .142    573 36661  GET /?Gq4KZlvaQRv0dzks=jpCEBcAI&KL0fr=2YE&ROeCOqswpGPRsJYtIEx=1LBPWiC HTTP/1.1
2022-02-08 13:17:48.999000 218.244.147.59  HTTP               .142    573 36822  GET /?Gq4KZlvaQRv0dzks=jpCEBcAI&KL0fr=2YE&ROeCOqswpGPRsJYtIEx=1LBPWiC HTTP/1.1
2022-02-08 13:17:49.098000 218.244.147.59  HTTP               .142    596 36718  HEAD /?magBXfv0cEMPdOul=Tssee7aX&w00d6=DVh&bJ6vWhBzdM2qnRaLhOL=DVgiIY0 HTTP/1.1
2022-02-08 13:17:49.297000 218.244.147.59  HTTP               .142    645 36489  POST /?CQEJo2YC6ZIuIBa9=FE59JE2T&5QGtC=rJ7&7cLuakqUddYiVFiSr7V=Gb1U7mF HTTP/1.1
2022-02-08 13:17:49.332000 218.244.147.59  HTTP               .142    646 36825  DELETE /?0UwO6GMKCkw5UG5i=gJF15NND&FV3lZ=5oW&IbbcUtP0rX8gDaNZNee=7NI0axV HTTP/1.1
2022-02-08 13:17:49.466000 218.244.147.59  HTTP               .142    574 36827  HEAD /?udHXmcGy1sGQsW7H=uPvCKnEI&qbQdx=50t&tPlFHtGUkYPwCvgmWEe=u7Ubt4L HTTP/1.1
2022-02-08 13:17:49.510000 218.244.147.59  HTTP               .142    596 36828  HEAD /?magBXfv0cEMPdOul=Tssee7aX&w00d6=DVh&bJ6vWhBzdM2qnRaLhOL=DVgiIY0 HTTP/1.1
2022-02-08 13:17:49.693000 218.244.147.59  HTTP               .142    645 36833  POST /?CQEJo2YC6ZIuIBa9=FE59JE2T&5QGtC=rJ7&7cLuakqUddYiVFiSr7V=Gb1U7mF HTTP/1.1
2022-02-08 13:17:50.076000 218.244.147.59  HTTP               .142    587 36709  DELETE /?Gn6ypxKhz15avPdL=T6osDWsB&9TWet=sNk&e4qOwCzQSZBkgqvZKpG=FeZrs9Q HTTP/1.1
2022-02-08 13:17:50.420000 218.244.147.59  HTTP               .142    587 36846  DELETE /?Gn6ypxKhz15avPdL=T6osDWsB&9TWet=sNk&e4qOwCzQSZBkgqvZKpG=FeZrs9Q HTTP/1.1
```

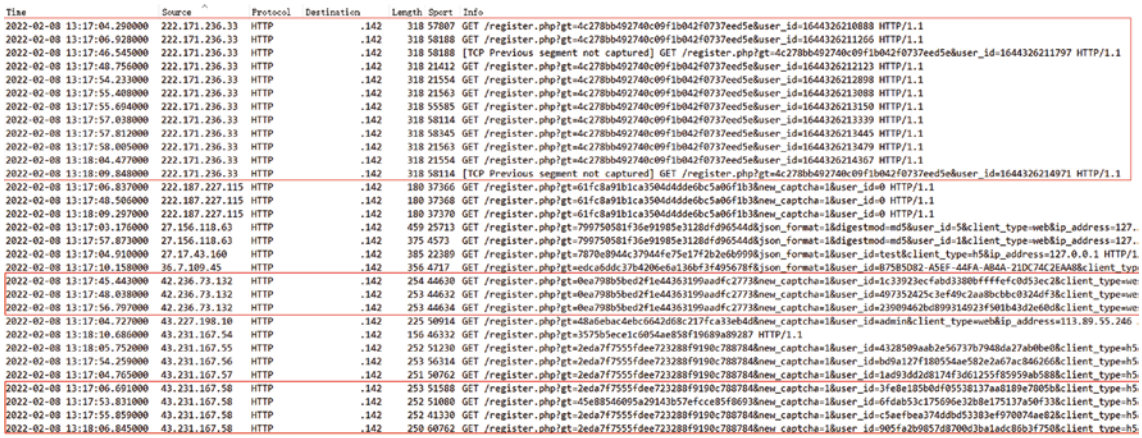
HTTP methods supported by APIs include GET, POST, DELETE, PUT, and HEAD, all of which can be used to launch attacks. Unlike websites, APIs do not provide direct access to the root directory. An attacker attempts to use a variable HTTP method to initiate a root directory request to the API server, consuming the server performance.

» **Using fixed API requests with illegitimate parameters to attack specific APIs**

Attackers select the API with the longest response time as the attack target in advance and use random parameters to consume more database query performance and improve the attack effect.

Registration APIs cause more content checks performed by the server, which is favored by attackers.

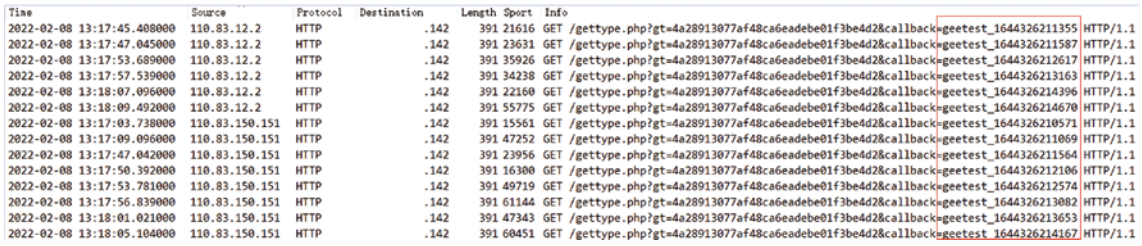
Obtained Packets of a Registration API Attack Using GET Request



Time	Source	Protocol	Destination	Length	Sport	Info
2022-02-08 13:17:04.290000	222.171.236.33	HTTP	.142	318	57807	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326210888 HTTP/1.1
2022-02-08 13:17:06.928000	222.171.236.33	HTTP	.142	318	58188	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326211266 HTTP/1.1
2022-02-08 13:17:46.545000	222.171.236.33	HTTP	.142	318	58188	[TCP Previous segment not captured] GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326211797 HTTP/1.1
2022-02-08 13:17:48.756000	222.171.236.33	HTTP	.142	318	21412	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326212123 HTTP/1.1
2022-02-08 13:17:54.233000	222.171.236.33	HTTP	.142	318	21554	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326212898 HTTP/1.1
2022-02-08 13:17:55.408000	222.171.236.33	HTTP	.142	318	21563	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326213088 HTTP/1.1
2022-02-08 13:17:55.694000	222.171.236.33	HTTP	.142	318	55585	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326213150 HTTP/1.1
2022-02-08 13:17:57.038000	222.171.236.33	HTTP	.142	318	58114	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326213339 HTTP/1.1
2022-02-08 13:17:57.812000	222.171.236.33	HTTP	.142	318	58345	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326213445 HTTP/1.1
2022-02-08 13:17:58.005000	222.171.236.33	HTTP	.142	318	21563	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326213479 HTTP/1.1
2022-02-08 13:18:04.477000	222.171.236.33	HTTP	.142	318	21554	GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326214367 HTTP/1.1
2022-02-08 13:18:09.848000	222.171.236.33	HTTP	.142	318	58114	[TCP Previous segment not captured] GET /register.php?gt=4c278bb492740c09f1b042f0737eed5e&user_id=1644326214971 HTTP/1.1
2022-02-08 13:17:06.837000	222.187.227.115	HTTP	.142	180	37366	GET /register.php?gt=61fc8a91b1ca3504d4dde6bc5a06f1b3&new_captcha=1&user_id=0 HTTP/1.1
2022-02-08 13:17:48.506000	222.187.227.115	HTTP	.142	180	37368	GET /register.php?gt=61fc8a91b1ca3504d4dde6bc5a06f1b3&new_captcha=1&user_id=0 HTTP/1.1
2022-02-08 13:18:09.297000	222.187.227.115	HTTP	.142	180	37370	GET /register.php?gt=61fc8a91b1ca3504d4dde6bc5a06f1b3&new_captcha=1&user_id=0 HTTP/1.1
2022-02-08 13:17:03.176000	27.156.118.63	HTTP	.142	459	25713	GET /register.php?gt=799750581f36e91985e3128dfd96544d&json_format=1&digestmod=md5&user_id=5&client_type=web&ip_address=127.
2022-02-08 13:17:57.873000	27.156.118.63	HTTP	.142	375	4573	GET /register.php?gt=799750581f36e91985e3128dfd96544d&json_format=1&digestmod=md5&user_id=1&client_type=web&ip_address=127.
2022-02-08 13:17:04.910000	27.17.43.160	HTTP	.142	385	22389	GET /register.php?gt=7870e8944c37944fe75e17f2b2e6b999&json_format=1&user_id=test&client_type=h5&ip_address=127.0.0.1 HTTP/1
2022-02-08 13:17:10.158000	36.7.109.45	HTTP	.142	356	4717	GET /register.php?gt=edca6ddc37b4206e6a136bf3f495678f&json_format=1&user_id=B75B5D82-A5EF-44FA-A04A-21DC74C2EAA8&client_typ
2022-02-08 13:17:45.443000	42.236.73.132	HTTP	.142	254	44630	GET /register.php?gt=0ea798b5bed2f1e44363199aadfc2773&new_captcha=1&user_id=1c33923ecfabd3380bfffefc0d53ec2&client_type=we
2022-02-08 13:17:48.038000	42.236.73.132	HTTP	.142	253	44632	GET /register.php?gt=0ea798b5bed2f1e44363199aadfc2773&new_captcha=1&user_id=497352425c3ef49c2aa8bcbbc0324df3&client_type=we
2022-02-08 13:17:56.797000	42.236.73.132	HTTP	.142	253	44634	GET /register.php?gt=0ea798b5bed2f1e44363199aadfc2773&new_captcha=1&user_id=23909462bd899314923f501b43d2e60d&client_type=we
2022-02-08 13:17:04.727000	43.227.198.10	HTTP	.142	225	50914	GET /register.php?gt=48a6ebac4ebc6642d68c217fca33eb4d&new_captcha=1&user_id=admin&client_type=web&ip_address=113.89.55.246
2022-02-08 13:18:10.686000	43.231.167.54	HTTP	.142	156	46332	GET /register.php?gt=3575b5ece1c6054ae858f19689a89287 HTTP/1.1
2022-02-08 13:18:05.752000	43.231.167.55	HTTP	.142	252	51230	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=4328509aab2e56737b7948da27ab0be0&client_type=h5
2022-02-08 13:17:54.259000	43.231.167.56	HTTP	.142	253	56314	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=bd9a127f180554ae582e2a67ac846266&client_type=h5
2022-02-08 13:17:04.765000	43.231.167.57	HTTP	.142	251	50762	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=1ad93dd2d8174f3d61255f85959ab588&client_type=h5
2022-02-08 13:17:06.691000	43.231.167.58	HTTP	.142	253	51588	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=3fe8e185b0df05538137aa8189e7005b&client_type=h5
2022-02-08 13:17:53.831000	43.231.167.58	HTTP	.142	252	51080	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=6fdab53c175696e32b8e175137a50f33&client_type=h5
2022-02-08 13:17:55.859000	43.231.167.58	HTTP	.142	252	41330	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=c5aefbea374ddbd53383ef970074ae82&client_type=h5
2022-02-08 13:18:06.845000	43.231.167.58	HTTP	.142	250	60762	GET /register.php?gt=2eda7f7555fdee723288f9190c788784&new_captcha=1&user_id=905fa2b9857d8700d3ba1adc85b3f750&client_type=h5

When an attacker selects a query API as the target, the query parameters change randomly to improve the attack effect.

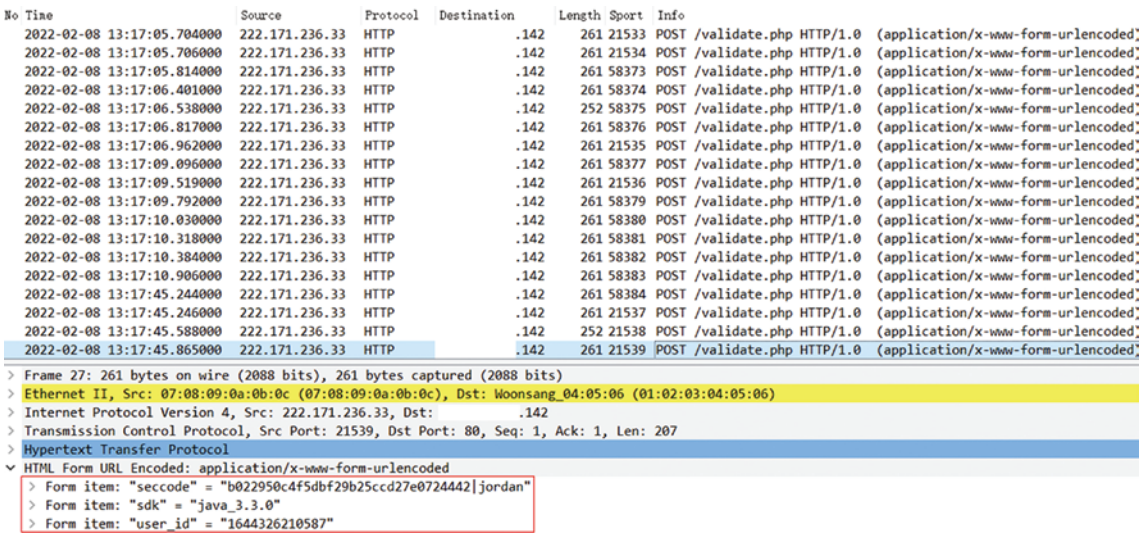
Obtained Packets of a Query API Attack Using GET Request



Time	Source	Protocol	Destination	Length	Sport	Info
2022-02-08 13:17:45.408000	110.83.12.2	HTTP	.142	391	21616	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326211355 HTTP/1.1
2022-02-08 13:17:47.045000	110.83.12.2	HTTP	.142	391	23631	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326211587 HTTP/1.1
2022-02-08 13:17:53.689000	110.83.12.2	HTTP	.142	391	35926	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326212617 HTTP/1.1
2022-02-08 13:17:57.539000	110.83.12.2	HTTP	.142	391	34238	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326213163 HTTP/1.1
2022-02-08 13:18:07.096000	110.83.12.2	HTTP	.142	391	22160	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326214396 HTTP/1.1
2022-02-08 13:18:09.492000	110.83.12.2	HTTP	.142	391	55775	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326214670 HTTP/1.1
2022-02-08 13:17:03.738000	110.83.150.151	HTTP	.142	391	15561	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326210571 HTTP/1.1
2022-02-08 13:17:09.096000	110.83.150.151	HTTP	.142	391	47252	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326211069 HTTP/1.1
2022-02-08 13:17:47.042000	110.83.150.151	HTTP	.142	391	23956	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326211564 HTTP/1.1
2022-02-08 13:17:50.392000	110.83.150.151	HTTP	.142	391	16300	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326212106 HTTP/1.1
2022-02-08 13:17:53.781000	110.83.150.151	HTTP	.142	391	49719	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326212574 HTTP/1.1
2022-02-08 13:17:56.839000	110.83.150.151	HTTP	.142	391	61144	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326213082 HTTP/1.1
2022-02-08 13:18:01.021000	110.83.150.151	HTTP	.142	391	47343	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326213653 HTTP/1.1
2022-02-08 13:18:05.104000	110.83.150.151	HTTP	.142	391	60451	GET /gettype.php?gt=4a28913077af48ca6eadebe01f3be4d2&callback=geetest_1644326214167 HTTP/1.1

APIs for permission verification are also vulnerable to attacks. To improve the attack effect, form content submitted by POST changes randomly.

Obtained Packets of a Validation API Attack Using POST Request



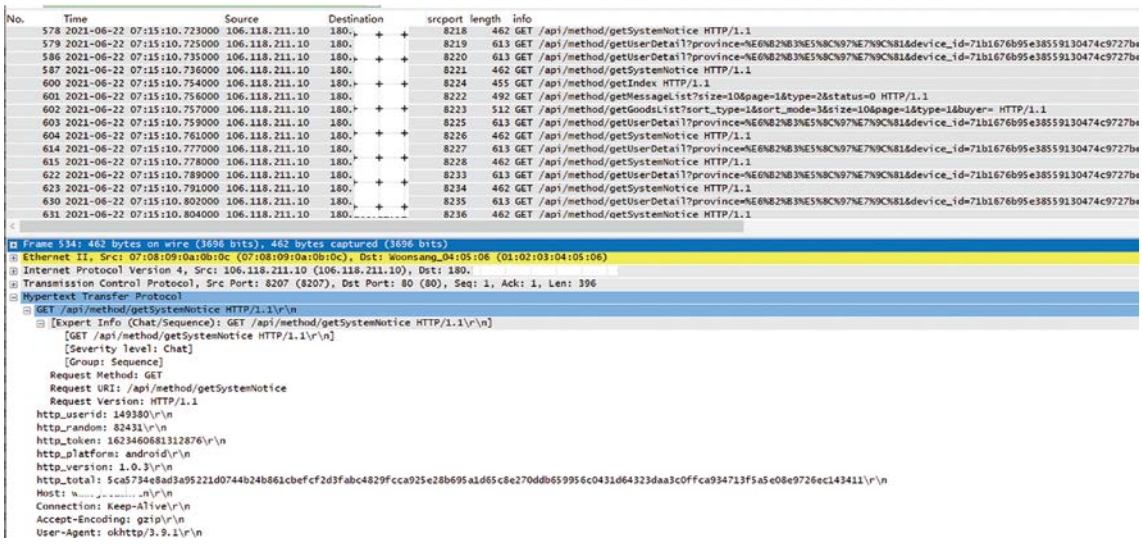
No	Time	Source	Protocol	Destination	Length	Sport	Info
	2022-02-08 13:17:05.704000	222.171.236.33	HTTP	.142	261	21533	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:05.706000	222.171.236.33	HTTP	.142	261	21534	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:05.814000	222.171.236.33	HTTP	.142	261	58373	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:06.401000	222.171.236.33	HTTP	.142	261	58374	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:06.538000	222.171.236.33	HTTP	.142	252	58375	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:06.817000	222.171.236.33	HTTP	.142	261	58376	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:06.962000	222.171.236.33	HTTP	.142	261	21535	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:09.096000	222.171.236.33	HTTP	.142	261	58377	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:09.519000	222.171.236.33	HTTP	.142	261	21536	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:09.792000	222.171.236.33	HTTP	.142	261	58379	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:10.030000	222.171.236.33	HTTP	.142	261	58380	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:10.318000	222.171.236.33	HTTP	.142	261	58381	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:10.384000	222.171.236.33	HTTP	.142	261	58382	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:10.906000	222.171.236.33	HTTP	.142	261	58383	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:45.244000	222.171.236.33	HTTP	.142	261	58384	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:45.246000	222.171.236.33	HTTP	.142	261	21537	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:45.588000	222.171.236.33	HTTP	.142	252	21538	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)
	2022-02-08 13:17:45.865000	222.171.236.33	HTTP	.142	261	21539	POST /validate.php HTTP/1.0 (application/x-www-form-urlencoded)

```
> Frame 27: 261 bytes on wire (2088 bits), 261 bytes captured (2088 bits)
> Ethernet II, Src: 07:08:09:0a:0b:0c (07:08:09:0a:0b:0c), Dst: Woonsang_04:05:06 (01:02:03:04:05:06)
> Internet Protocol Version 4, Src: 222.171.236.33, Dst:           .142
> Transmission Control Protocol, Src Port: 21539, Dst Port: 80, Seq: 1, Ack: 1, Len: 207
> Hypertext Transfer Protocol
v HTML Form URL Encoded: application/x-www-form-urlencoded
  > Form item: "seccode" = "b022950c4f5dbf29b25ccd27e0724442|jordan"
  > Form item: "sdk" = "java_3.3.0"
  > Form item: "user_id" = "1644326210587"
```

» **Using variable API requests to evade data filtering**

Malicious actors attack multiple APIs in order to increase defense difficulties. If the API keyword of a request is randomly generated, the request is for a non-existent resource, which further deteriorates the API server performance.

Obtained Packets of a High-Rate Variable API Request Attack



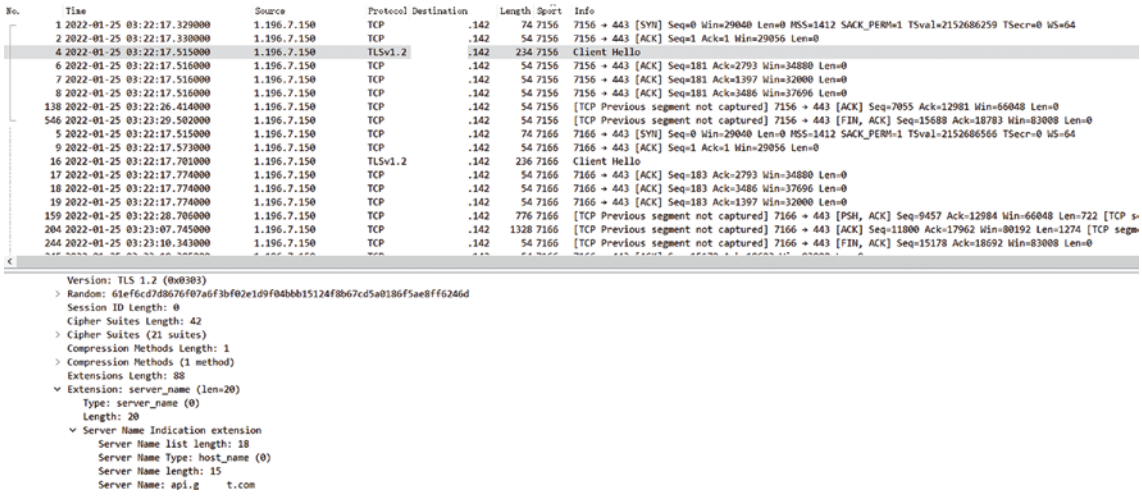
3. TLS Attacks

To improve security, APIs are encrypted using TLS. As a result, APIs are vulnerable to attacks at the TLS layer.

» **Incomplete TLS sessions: Sending only client hello packets**

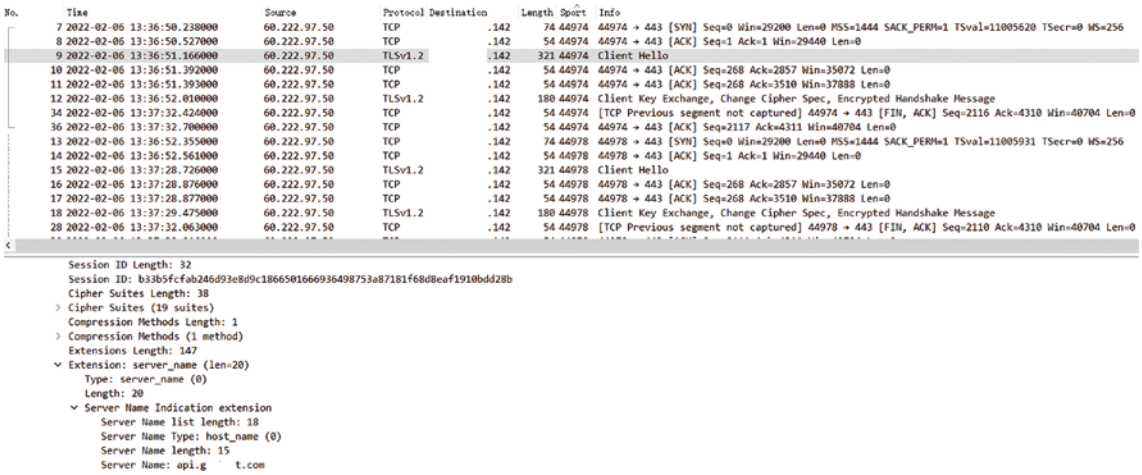
After establishing a TCP session with the API server, the attacker sends only client hello packets to disconnect the session, consuming TLS resources on the server.

Attacking an API Through Incomplete TLS Sessions: Only Client Hello Packets



» **Incomplete TLS sessions: Complete TLS handshake but no data exchange**
During the TLS handshake, the server performance consumption is 15 times that of the client. Therefore, attackers often establish TCP sessions with the API server and complete the TLS handshake to improve the attack effect.

Attacking an API Through Incomplete TLS Sessions: Complete TLS Handshake and No Data Exchange



No.	Time	Source	Protocol	Destination	Length	Sport	Info
7	2022-02-06 13:36:50.238000	60.222.97.50	TCP	.142	74	44974	44974 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1444 SACK_PERM=1 TSval=11005620 TSecr=0 WS=256
8	2022-02-06 13:36:50.527000	60.222.97.50	TCP	.142	54	44974	44974 → 443 [ACK] Seq=1 Ack=1 Win=29440 Len=0
9	2022-02-06 13:36:51.166000	60.222.97.50	TLSv1.2	.142	321	44974	Client Hello
10	2022-02-06 13:36:51.392000	60.222.97.50	TCP	.142	54	44974	44974 → 443 [ACK] Seq=268 Ack=2857 Win=35072 Len=0
11	2022-02-06 13:36:51.393000	60.222.97.50	TCP	.142	54	44974	44974 → 443 [ACK] Seq=268 Ack=3510 Win=37888 Len=0
12	2022-02-06 13:36:52.010000	60.222.97.50	TLSv1.2	.142	180	44974	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
34	2022-02-06 13:37:32.424000	60.222.97.50	TCP	.142	54	44974	[TCP Previous segment not captured] 44974 → 443 [FIN, ACK] Seq=2116 Ack=4310 Win=40704 Len=0
36	2022-02-06 13:37:32.700000	60.222.97.50	TCP	.142	54	44974	44974 → 443 [ACK] Seq=2117 Ack=4311 Win=40704 Len=0
13	2022-02-06 13:36:52.355000	60.222.97.50	TCP	.142	74	44978	44978 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1444 SACK_PERM=1 TSval=11005931 TSecr=0 WS=256
14	2022-02-06 13:36:52.561000	60.222.97.50	TCP	.142	54	44978	44978 → 443 [ACK] Seq=1 Ack=1 Win=29440 Len=0
15	2022-02-06 13:37:28.726000	60.222.97.50	TLSv1.2	.142	321	44978	Client Hello
16	2022-02-06 13:37:28.876000	60.222.97.50	TCP	.142	54	44978	44978 → 443 [ACK] Seq=268 Ack=2857 Win=35072 Len=0
17	2022-02-06 13:37:28.877000	60.222.97.50	TCP	.142	54	44978	44978 → 443 [ACK] Seq=268 Ack=3510 Win=37888 Len=0
18	2022-02-06 13:37:29.475000	60.222.97.50	TLSv1.2	.142	180	44978	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
28	2022-02-06 13:37:32.063000	60.222.97.50	TCP	.142	54	44978	[TCP Previous segment not captured] 44978 → 443 [FIN, ACK] Seq=2110 Ack=4310 Win=40704 Len=0

```
Session ID Length: 32
Session ID: b33b5fcfab246d93e8d9c18665016669364987 53a87181f68d8eaf1910bdd28b
Cipher Suites Length: 38
> Cipher Suites (19 suites)
Compression Methods Length: 1
> Compression Methods (1 method)
Extensions Length: 147
v Extension: server_name (len=20)
    Type: server_name (0)
    Length: 20
  v Server Name Indication extension
      Server Name list length: 18
      Server Name Type: host_name (0)
      Server Name length: 15
      Server Name: api.g      t.com
```

4. HTTP Application-Layer Attacks

HTTPS application-layer attacks targeting APIs can also be classified into fixed resource request attacks and variable resource request attacks.

» **Using fixed resource request attacks to consume server resources**
Attackers select the API with the longest response time as the attack target in advance. This type of encryption attack belongs to fixed resource request attack.

Attacking an Encrypted API Using Fixed Resource Requests

Time	Source	Protocol	Destination	Length	Sport	Info
2022-02-06 13:36:46.532000	113.101.44.132	TLSv1.2	.142	542	44152	Application Data
2022-02-06 13:37:49.650000	113.101.44.132	TLSv1.2	.142	542	42714	Application Data
2022-02-06 13:37:50.197000	113.101.44.132	TLSv1.2	.142	542	42567	Application Data
2022-02-06 13:37:50.983000	113.101.44.132	TLSv1.2	.142	542	43506	Application Data
2022-02-06 13:37:53.025000	113.101.44.132	TLSv1.2	.142	542	42652	Application Data
2022-02-06 13:36:48.370000	113.101.44.155	TLSv1.2	.142	542	27375	Application Data
2022-02-06 13:36:44.636000	113.101.44.33	TLSv1.2	.142	542	40479	Application Data
2022-02-06 13:36:45.068000	113.101.44.33	TLSv1.2	.142	542	41923	Application Data
2022-02-06 13:36:45.228000	113.101.44.33	TLSv1.2	.142	542	39506	Application Data
2022-02-06 13:36:51.267000	113.101.45.197	TLSv1.2	.142	542	33057	Application Data
2022-02-06 13:36:44.643000	113.101.45.233	TLSv1.2	.142	542	2916	Application Data
2022-02-06 13:36:44.643000	113.101.45.233	TLSv1.2	.142	542	4714	Application Data
2022-02-06 13:36:45.972000	113.101.45.233	TLSv1.2	.142	542	2705	Application Data
2022-02-06 13:37:52.499000	113.101.45.233	TLSv1.2	.142	542	2134	Application Data
2022-02-06 13:37:45.486000	113.103.112.57	TLSv1.2	.142	542	28339	Application Data
2022-02-06 13:36:45.694000	113.103.113.251	TLSv1.2	.142	542	25402	Application Data
2022-02-06 13:36:52.509000	113.103.113.251	TLSv1.2	.142	542	25402	Application Data
2022-02-06 13:37:30.005000	113.103.113.251	TLSv1.2	.142	542	21527	Application Data
2022-02-06 13:37:32.329000	113.103.113.251	TLSv1.2	.142	542	21562	Application Data
2022-02-06 13:37:42.637000	113.103.113.251	TLSv1.2	.142	542	24136	Application Data
2022-02-06 13:36:44.786000	113.103.115.34	TLSv1.2	.142	542	17191	Application Data
2022-02-06 13:37:48.331000	113.103.115.34	TLSv1.2	.142	542	14144	Application Data
2022-02-06 13:36:52.805000	113.103.115.46	TLSv1.2	.142	542	7214	Application Data
2022-02-06 13:37:30.368000	113.103.115.46	TLSv1.2	.142	542	7490	Application Data
2022-02-06 13:36:45.080000	113.110.33.110	TLSv1.2	.142	542	57427	Application Data
2022-02-06 13:36:45.525000	113.110.33.110	TLSv1.2	.142	542	58005	Application Data
2022-02-06 13:36:45.682000	113.110.33.110	TLSv1.2	.142	542	57768	Application Data
2022-02-06 13:36:49.274000	113.110.33.110	TLSv1.2	.142	542	54433	Application Data
2022-02-06 13:36:50.041000	113.110.33.110	TLSv1.2	.142	542	54443	Application Data
2022-02-06 13:36:50.231000	113.110.33.110	TLSv1.2	.142	542	54449	Application Data
2022-02-06 13:36:50.940000	113.110.33.110	TLSv1.2	.142	542	55320	Application Data
2022-02-06 13:36:47.013000	113.110.33.141	TLSv1.2	.142	542	27457	Application Data
2022-02-06 13:37:46.097000	113.110.33.141	TLSv1.2	.142	542	28147	Application Data
2022-02-06 13:37:46.499000	113.110.33.141	TLSv1.2	.142	542	28153	Application Data
2022-02-06 13:37:47.084000	113.110.33.141	TLSv1.2	.142	542	28163	Application Data

» **Using variable resource request attacks to evade defense**

When an attacker attacks multiple APIs, this type of encryption attack belongs to variable resource request attack. When the security system performs defense actions after decryption, it is difficult to defend against one source IP address accessing multiple URLs concurrently.

Attacking an Encrypted API Using Variable Resource Requests

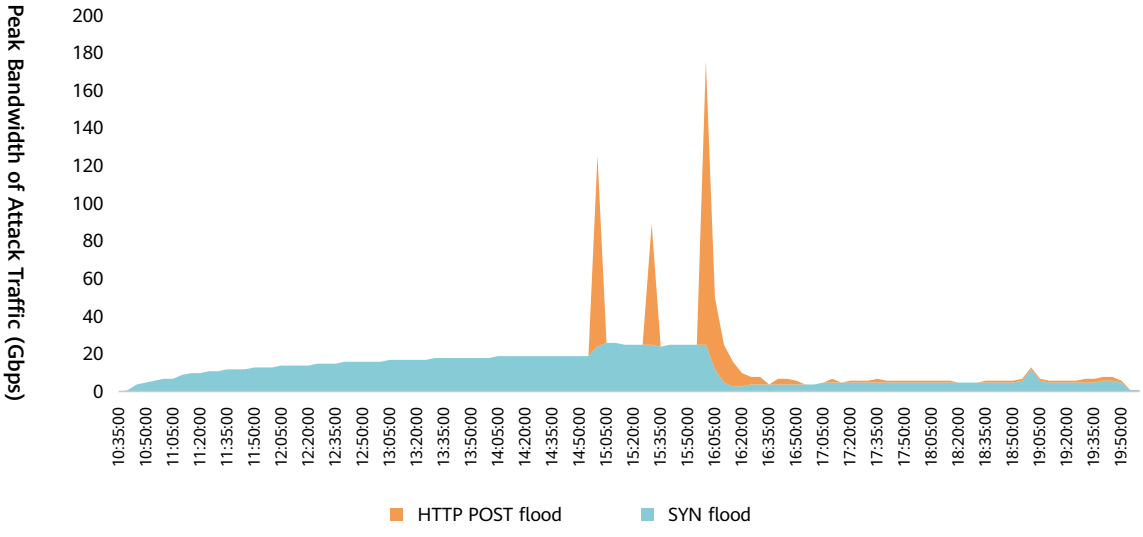
Time	Source	Protocol	Destination	Length	Sport	Info
2022-02-06 13:37:46.536000	106.224.91.54	TLSv1.2	.142	386	22216	Application Data
2022-02-06 13:37:46.659000	106.224.91.54	TLSv1.2	.142	386	22217	Application Data
2022-02-06 13:37:47.639000	106.224.91.54	TLSv1.2	.142	763	22218	Application Data
2022-02-06 13:37:47.931000	106.224.91.54	TLSv1.2	.142	386	22219	Application Data
2022-02-06 13:37:44.621000	113.121.23.114	TLSv1.2	.142	378	42619	Application Data
2022-02-06 13:37:46.817000	113.121.23.114	TLSv1.1	.142	395	42644	Application Data
2022-02-06 13:37:50.078000	113.121.23.114	TLSv1.2	.142	378	42682	Application Data
2022-02-06 13:37:31.181000	113.218.232.219	TLSv1.2	.142	602	52077	Application Data
2022-02-06 13:37:50.910000	113.218.232.219	TLSv1.2	.142	697	52099	Application Data
2022-02-06 13:37:52.119000	113.218.232.219	TLSv1.2	.142	917	52101	Application Data
2022-02-06 13:37:30.009000	114.99.196.26	TLSv1.2	.142	378	54256	Application Data
2022-02-06 13:37:35.522000	114.99.196.26	TLSv1	.142	1328	53835	Application Data
2022-02-06 13:37:38.533000	114.99.196.26	TLSv1.2	.142	378	54828	Application Data
2022-02-06 13:37:47.227000	114.99.196.26	TLSv1	.142	1328	55277	Application Data
2022-02-06 13:37:47.746000	114.99.196.26	TLSv1.2	.142	378	55313	Application Data
2022-02-06 13:37:50.045000	114.99.196.26	TLSv1	.142	1328	55433	Application Data
2022-02-06 13:37:52.589000	114.99.196.26	TLSv1.2	.142	378	55520	Application Data
2022-02-06 13:37:30.876000	121.226.11.27	TLSv1	.142	416	59221	Application Data, Application Data
2022-02-06 13:37:42.190000	121.226.11.27	TLSv1.2	.142	623	51077	Application Data
2022-02-06 13:37:47.046000	121.226.11.27	TLSv1.2	.142	378	54940	Application Data
2022-02-06 13:37:49.711000	121.226.11.27	TLSv1.2	.142	378	56419	Application Data
2022-02-06 13:36:46.106000	140.250.148.187	TLSv1	.142	416	48034	Application Data, Application Data
2022-02-06 13:37:31.823000	140.250.148.187	TLSv1.2	.142	378	48134	Application Data
2022-02-06 13:37:39.763000	140.250.148.187	TLSv1.2	.142	378	48180	Application Data
2022-02-06 13:37:50.169000	140.250.148.187	TLSv1	.142	416	48255	Application Data, Application Data
2022-02-06 13:37:32.417000	140.250.94.14	TLSv1	.142	416	53446	Application Data, Application Data
2022-02-06 13:37:33.039000	140.250.94.14	TLSv1.2	.142	378	53449	Application Data
2022-02-06 13:37:37.884000	140.250.94.14	TLSv1	.142	1328	53473	Application Data
2022-02-06 13:37:41.655000	171.114.188.199	TLSv1.2	.142	769	57194	Application Data
2022-02-06 13:37:49.768000	171.114.188.199	TLSv1.2	.142	767	54637	Application Data
2022-02-06 13:37:51.006000	171.114.188.199	TLSv1.2	.142	905	56881	Application Data

4.1.3 Typical Attack Incidents

1. Analysis on High-Intensity CC Attacks Targeting Ad APIs

In March 2019, Huawei detected a large-scale DDoS attack on the ad APIs. The attack consists of the real source SYN flood and HTTP POST flood and lasts for 9 hours and 15 minutes. The peak bandwidth of the attack traffic reached 175 Gbps. The attack comes from smartphones with the peak request rate reaching 17.4 Mrps, and up to 10 million smartphones were involved in the attack.

Traffic Distribution of Smartphone-Initiated Attacks on Ad APIs



According to the captured packets, the zombies used in the attack are mainly mobile phones of a single brand.

Obtained Packets of HTTP POST Flood on an Ad API

No.	Time	Source	Protocol	Destination	Length	Sport	Info
	2019-03-19 06:11:10.147000	106.108.1.228	HTTP	.71	1255	8942	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.177000	60.188.255.60	HTTP	.71	1237	35513	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.182000	182.134.62.49	HTTP	.71	1347	38953	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.185000	182.134.62.49	HTTP	.71	1347	47490	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.192000	223.104.16.80	HTTP	.71	1249	36687	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.209000	116.11.175.179	HTTP	.71	1201	1882	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.229000	27.195.202.125	HTTP	.71	1236	36911	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.244000	27.186.184.206	HTTP	.71	1237	4320	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.246000	125.93.149.198	HTTP	.71	1210	49186	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.251000	120.40.209.174	HTTP	.71	1334	48719	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.257000	139.207.75.40	HTTP	.71	1248	18295	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.270000	223.104.3.191	HTTP	.71	1232	12270	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.294000	58.54.84.107	HTTP	.71	1236	32878	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)
	2019-03-19 06:11:10.302000	171.104.16.36	HTTP	.71	1258	3690	POST /index/checkupdateencrypt HTTP/1.1 (application/x-www-form-urlencoded)

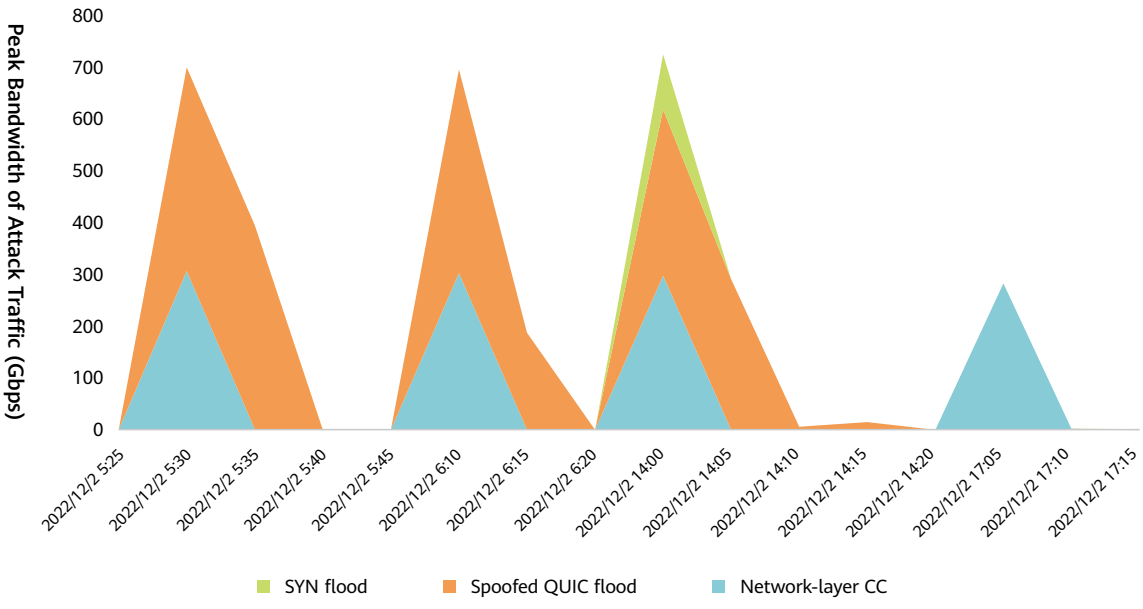
```
> Frame 10: 1255 bytes on wire (10040 bits), 1255 bytes captured (10040 bits)
> Ethernet II, Src: 07:08:09:0a:0b:0c (07:08:09:0a:0b:0c), Dst: Woonsang_04:05:06 (01:02:03:04:05:06)
> Internet Protocol Version 4, Src: 106.108.1.228, Dst:        .71
> Transmission Control Protocol, Src Port: 8942, Dst Port: 80, Seq: 1, Ack: 1, Len: 1201
v Hypertext Transfer Protocol
  > POST /index/checkupdateencrypt HTTP/1.1\r\n
    Content-Type: application/x-www-form-urlencoded\r\n
    User-Agent: Dalvik/2.1.0 (Linux; U; Android 7.1.1; GIONEE S10C Build/NMF26F)\r\n
    Host: sdkinapi.      .  .com\r\n
    Connection: Keep-Alive\r\n
    Accept-Encoding: gzip\r\n
  > Content-Length: 931\r\n
    \r\n
```

2. Analysis on Continuous Attacks Targeting Payment APIs

During the 2022 World Cup, Huawei detected that payment APIs suffered continuous and high-intensity DDoS attacks. The peak attack bandwidth exceeded 500 Gbps for three times, which were 698 Gbps, 694 Gbps, and 723 Gbps.

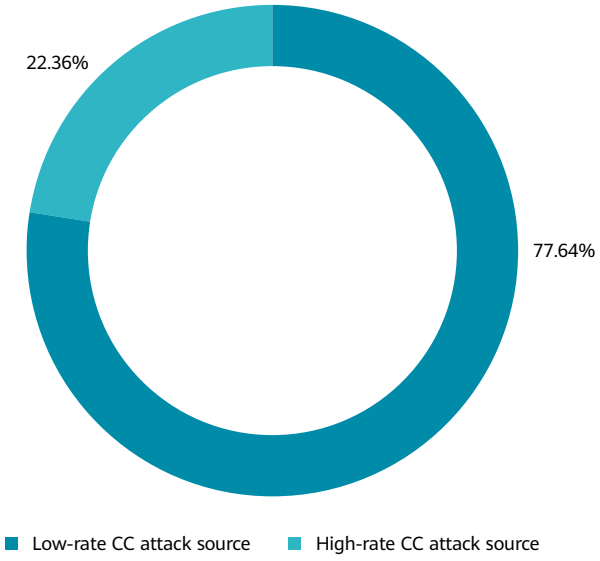
Attacks mainly consist of SYN flood, network-layer CC, and spoofed QUIC flood attacks, lasting for 11 hours and 50 minutes.

Traffic Distribution of Attacks on Payment APIs During the 2022 World Cup



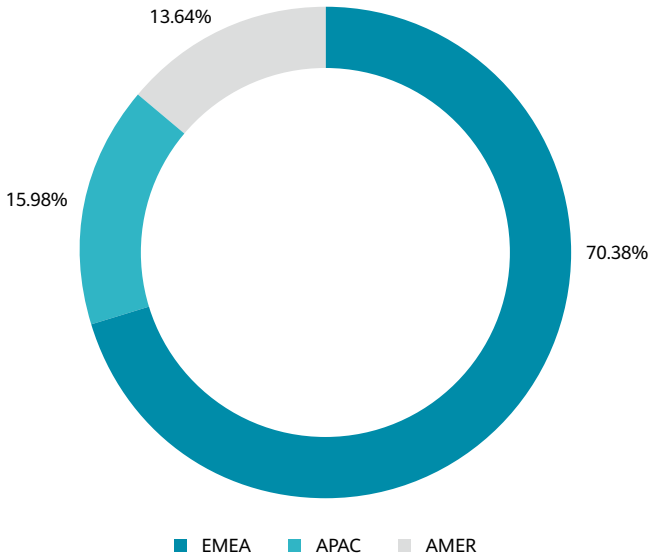
The network-layer CC attack uses large ACK packets in combination of both low-rate and high-rate modes. The packet sending rate of a high-rate attack source and that of a low-rate attack source ranges from 1000 pps to 3000 pps, and from 100 pps to 1000 pps, respectively. In addition, low-rate attacks account for 77.64%.

Distribution of High-Rate and Low-Rate CC Attack Sources at the Network Layer



According to the regional distribution of zombies initiating network-layer CC attacks, 70.38% of the zombies are from EMEA.

Distribution of Network-layer CC Zombies by Regions



4.1.4 DDoS Mitigation Suggestions

1. Attack Summary

1) APIs are threatened by DDoS attacks at Layer 3 to Layer 7 of the OSI Model

DDoS attacks on APIs are classified based on the OSI model. DDoS attacks exist at all layers except the physical layer and data link layer.

DDoS attacks targeting APIs (OSI model)

No.	Layer	Attack Type	Attack Impact
7	Application layer	• HTTP flood (such as HTTP GET flood, HTTP POST flood, HTTP PUT flood, HTTP HEAD flood and HTTP DELETE flood) • HTTPS flood	Server resources
6	Presentation layer	TLS attack (incomplete TLS session)	Server resources
5	Session layer	Network-layer CC attack (HTTP null connection and TLS null connection)	Bandwidth resources Server performance
4	Transport layer	SYN flood UDP flood UDP reflection TCP reflection	Session resources Bandwidth resources
3	Network layer	UDP fragment flood TCP fragment flood ICMP flood	Bandwidth resources
2	Data link layer	N/A	N/A
1	Physical layer	N/A	N/A

2) Illegitimate access attacks are launched by requesting the root directory

Frequently requesting the root directory is a common tactic to attack APIs. Using the root directory to attack websites is a resource abuse caused by excessive access to legitimate resources. For API services, each API corresponds to a specific URL and the URL varies according to the API. The API does not support root directory access. Therefore, it is illegitimate to attack APIs by requesting the root directory.

3) Large resource attacks are generated by requesting huge forms

Websites and apps use a large number of images to improve customer experience. Therefore, large resource attacks caused by images account for a high proportion. APIs do not provide web UIs and therefore do not involve large resource attacks caused by large files such as images and upgrade packages. However, query APIs face large resource attacks caused by attackers' continuous requests for huge forms.

4) Abuse risks are increased by requesting multiple methods

The API provides data exchange between applications. It uses the GET method to obtain data, the PUT method to update data, the POST method to submit new data, and the DELETE method to delete data. As a result, the API has higher method abuse risks. In addition to common HTTP GET flood and HTTP POST flood attacks, HTTP PUT flood, HTTP DELETE flood and HTTP HEAD flood are also used to attack APIs.

2. Defense Suggestions

Compared with websites and apps, APIs often provide machine-machine interfaces and cannot block robot attacks using traditional source challenge authentication technologies (such as JavaScript authentication and CAPTCHA authentication) and mobile SDKs. Enterprises must mitigate DDoS attacks from multiple dimensions, such as API security architecture design, API application robustness design, and defense system construction.

1) Improving the overall security of APIs through security architecture design and mitigating DDoS threats

Many APIs do not have the load balancing + server cluster architecture, leading to insufficient performance. Therefore, application-layer attacks may pose a greater threat to APIs than to websites. For important API services, edge CDN acceleration + local load balancing + server cluster can be used to improve API processing performance and reduce application-layer attacks that consume server resources.

For important API services, attack threats can be narrowed down to traditional network-layer attacks and TLS attacks through TLS strong authentication, simplifying defense.

2) Enhancing the robustness of API programs and mitigating DDoS attacks introduced by illegitimate requests

Developers need to consider abnormal API invoking scenarios. For unsupported requests and non-compliant parameters, an exception handling process needs to be added and the error response code 40X needs to be displayed promptly, preventing attackers from crashing the API program through illegitimate requests.

3) Incorporating Internet-exposed APIs into the security protection system like websites

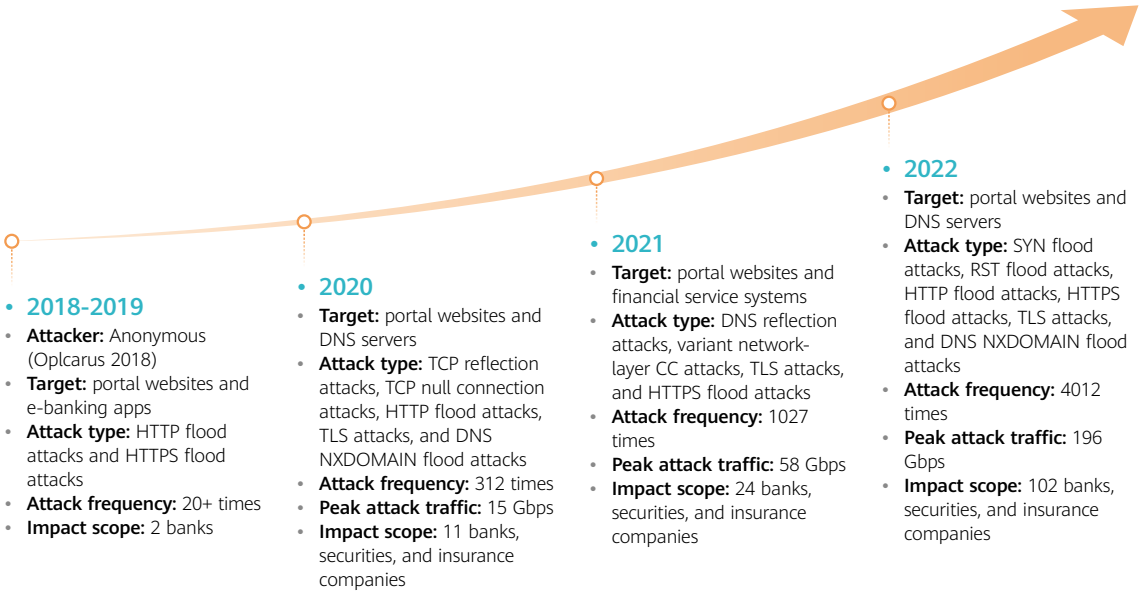
Enterprise security teams generally do not fully understand API security. As a result, risks of Internet-exposed APIs are ignored. Salt Security disclosed that 61% of enterprises lack basic API security policies[6].

Enterprises urgently need to identify the Internet-exposed APIs and incorporate such open APIs into the DDoS protection system. For example, the hierarchical defense architecture consisting of carrier upstream cloud mitigation + enterprise network border anti-DDoS + WAF/API gateway is used to ensure the API availability.

4.2 DDoS Attack Situation in the Finance Industry

Since 2015, the hacker organization named Anonymous has been launching large-scale DDoS attacks called OpIcarus targeting the finance industry. At least two Chinese financial enterprises were attacked by OpIcarus 2018. Since then, DDoS attacks against Chinese financial enterprises have become the norm, with portal websites, financial services systems (especially e-banking apps), and DNS servers becoming the main targets.

DDoS Attack Situation in China's Finance Industry

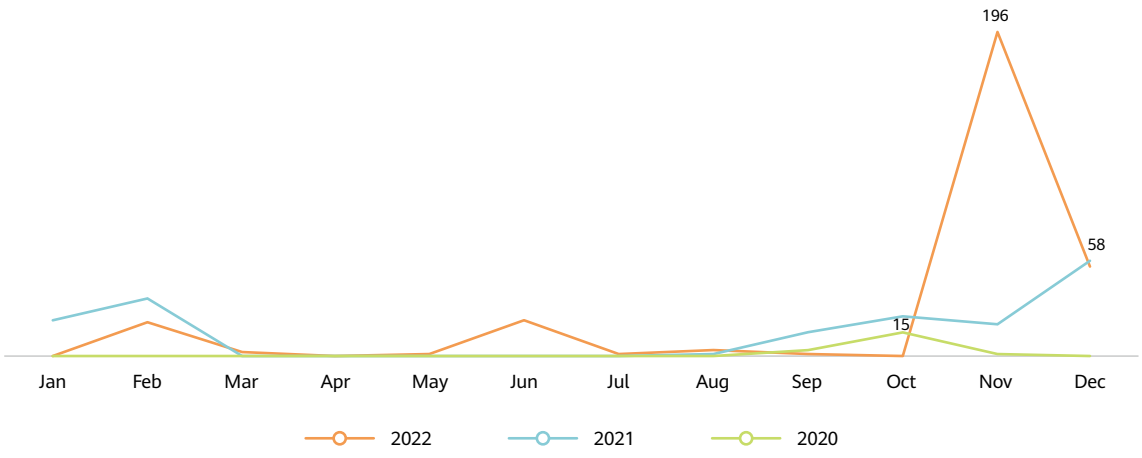


Source: China Unicom Digital Tech, Huawei

4.2.1 Attack Situation in the Finance Industry

1. Attack Intensity

Peak Attack Bandwidth of DDoS Attacks Targeting China's Finance Industry from 2020 to 2022 (Gbps)



Source: China Unicom Digital Tech, Huawei

From 2020 to 2022, the intensity of DDoS attacks on China's finance industry has greatly increased year on year. The peak attack bandwidth in 2021 was 3.8 times that in 2020, and the peak attack bandwidth in 2022 was 3.4 times that in 2021.

The maximum peak bandwidth of DDoS attacks in 2020 was 15 Gbps. It was generated during a 14-minute combo TCP reflection attack targeting a portal website in October that year.

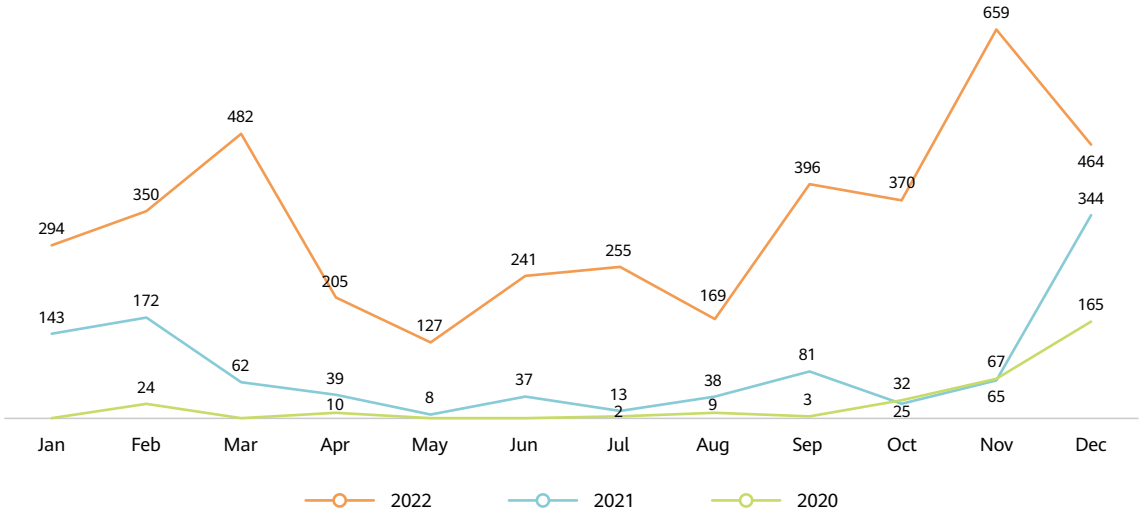
The maximum peak bandwidth of DDoS attacks in 2021 was 58 Gbps. It was generated during a 4-minute DNS reflection attack targeting a portal website in December that year.

The maximum peak bandwidth of DDoS attacks in 2022 was 196 Gbps. It was generated during 13-minute multi-vector attacks (including combo SYN flood, RST flood, HTTP flood, HTTPS flood, and TLS attacks) targeting a portal website in November that year.

In general, volumetric attacks usually occur during the service settlement period, and their peak bandwidth is far higher than the network link bandwidth of financial enterprises.

2. Attack Frequency

Attack Frequency of DDoS Attacks Targeting China's Finance Industry from 2020 to 2022



Source: China Unicom Digital Tech

From 2020 to 2022, the frequency of DDoS attacks targeting China's finance industry increased rapidly. In 2022, 4012 DDoS attacks occurred (3.9 times that in 2021 and 12.9 times that in 2020). Among them, most DDoS attacks occurred during the service settlement period, presenting obvious attack intents.

In 2022, a total of 102 banks, securities, and insurance enterprises were attacked by DDoS attacks (4.3 times that in 2021 and 9.3 times that in 2020).

In addition, DDoS attacks were launched on 2 large banks for 80 times in 2020, 6 large banks for 74 times in 2021, and 8 large banks for up to 193 times in 2022.

3. Attack Distribution

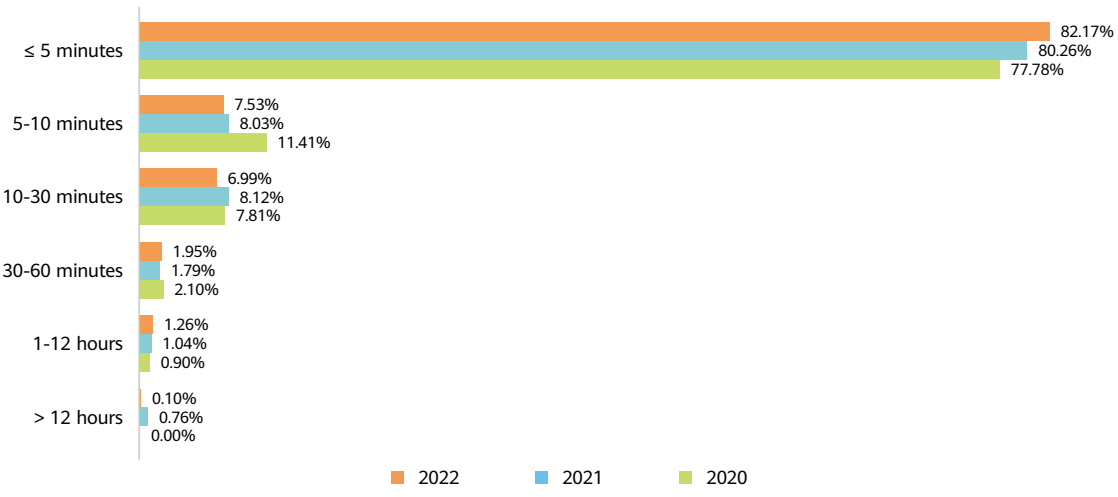
If DDoS attacks are classified based on the OSI model, DDoS threats exist on all layers except the physical layer and data link layer.

Classification of DDoS Attacks Targeting the Finance Industry According to the OSI Model

No.	Layer	Attack Type	Attack Impact
7	Application layer	• DNS NXDOMAIN flood • HTTP flood (HTTP GET flood, HTTP POST flood, HTTP HEAD flood, etc.) • HTTPS flood	Server resources Load balancing resources
6	Presentation layer	TLS attack (incomplete TLS session)	Load balancing resources
5	Session layer	Network-layer CC TCP null connection	Bandwidth resources Server resources TCP session resources
4	Transport layer	SYN flood RST flood SYN carpet-bombing UDP reflection TCP reflection UDP flood	Session resources Bandwidth resources
3	Network layer	UDP fragment flood ICMP flood	Bandwidth resources
2	Data link layer	N/A	N/A
1	Physical layer	N/A	N/A

4. Attack Duration

Attack Duration of DDoS Attacks Targeting China's Finance Industry from 2020 to 2022



Source: China Unicom Digital Tech

From 2020 to 2022, most DDoS attacks targeting the finance industry adopt "fast flooding" tactics, and mainly last for less than or equal to 5 minutes. In 2022, DDoS attacks with these features accounted for 82.17% of the total.

4.2.2 Analysis of Attacks Targeting the Finance Industry in 2018

In December 2018, Anonymous launched large-scale DDoS attacks named OpIcarus 2018 against global financial institutions with portal websites and mobile banking apps being the main targets. At least two Chinese financial enterprises were also hit by the attacks. 70% of the attack traffic came from outside China. In addition, a large number of IoT botnets were used by OpIcarus 2018 (mainly HTTP flood and HTTPS flood attacks), which lasted until mid-January 2019.

1. Attack technique analysis

» **High-rate large resource request attacks, consuming outbound network bandwidth**

To improve customer experience, a large number of images are used on the web UI of financial portal websites and mobile banking apps, and most of them are accelerated by CDN. To cause bandwidth congestion with a small number of access requests, hackers preferentially use the images that are not accelerated by CDN to attack financial enterprises.

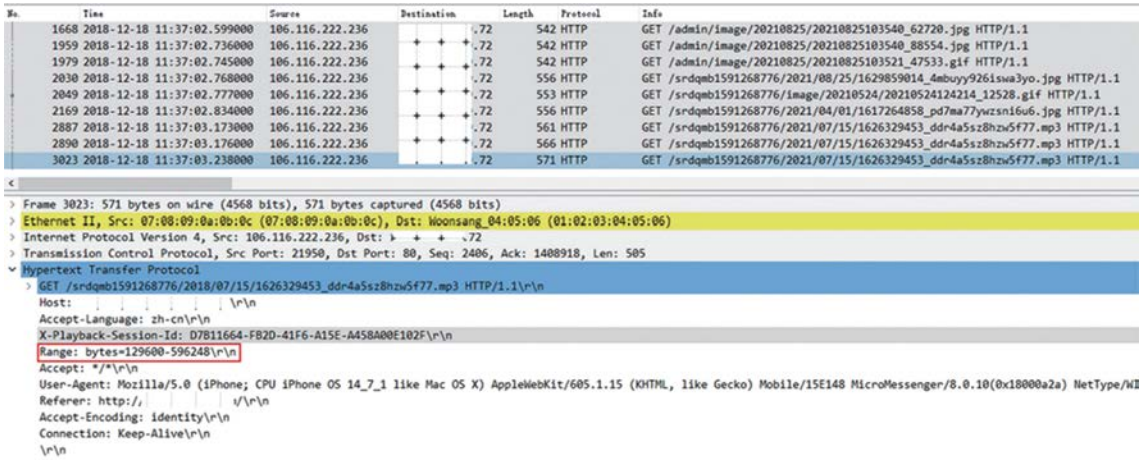
In 2018, hackers attacked financial portal websites and mobile banking apps using high-rate large resource request attacks.

Obtained Packets of a High-Rate Large Resource Request Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2018-12-17 01:33:30.092355	.11	HTTP	.34	790	28035	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:30.762628	.11	HTTP	.34	842	32962	GET /pweb/images/cardCredit_bg.png HTTP/1.1
2018-12-17 01:33:30.777157	.11	HTTP	.35	944	29035	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:30.972934	.11	HTTP	.35	931	29763	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:31.518225	.11	HTTP	.32	917	28633	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:32.271178	.11	HTTP	.32	882	27433	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:32.350382	.11	HTTP	.34	798	29141	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:33.471892	.11	HTTP	.32	844	28639	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:33.587047	.11	HTTP	.33	951	28009	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:34.136687	.11	HTTP	.35	854	29771	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:34.315313	.11	HTTP	.34	959	28621	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:35.023257	.11	HTTP	.35	901	33620	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:35.324092	.11	HTTP	.33	1095	34072	GET /pweb/images/cardCredit_bg.png HTTP/1.1
2018-12-17 01:33:35.408309	.11	HTTP	.35	939	29773	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:36.830176	.11	HTTP	.33	857	34076	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:37.712177	.11	HTTP	.34	930	32962	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:37.832994	.11	HTTP	.35	913	28315	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:37.981356	.11	HTTP	.33	844	32246	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:38.271846	.11	HTTP	.33	955	29289	GET /pweb/images/cardCredit_bg.png HTTP/1.1
2018-12-17 01:33:38.545162	.11	HTTP	.33	950	29289	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:38.600076	.11	HTTP	.34	849	34548	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:38.806179	.11	HTTP	.35	864	34512	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:38.939453	.11	HTTP	.35	800	29035	GET /pweb/images/card_bg.jpg HTTP/1.1
2018-12-17 01:33:39.291745	.11	HTTP	.34	951	33794	GET /pweb/images/card_bg.jpg HTTP/1.1

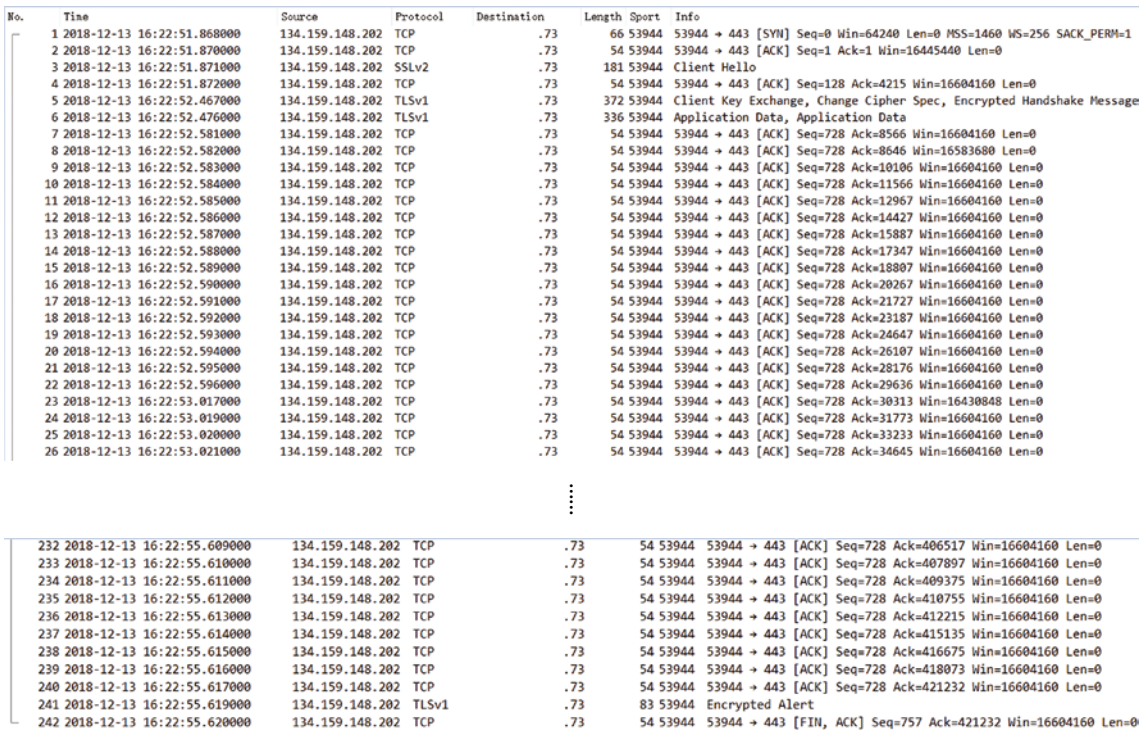
When an attacker launches high-rate large resource request attacks on some financial enterprises, the attacker sets the Range field in the HTTP header for repeated replies from the server. This further intensifies the attack effect on the outbound bandwidth.

Obtained Packets of a Range-based Amplification Attack



Generally, e-banking apps use HTTPS. The following figure shows the obtained attack packets of large resource request attacks targeting a mobile banking app. For a single session, an attacker sends a 336-byte request packet to trigger a response with a total length of 412,666 bytes, occupying an average outbound bandwidth of 1.5 Mbps.

Obtained Packets of a TLS Encrypted Large Resource Request Attack



As shown in the following figure, a single attack source can initiate a maximum of 15 requests per second, and the outbound traffic bandwidth occupied by the response (triggered by these requests) is about 22.5 Mbps. If an attacker invokes 1000 zombies to launch attacks, about 22 Gbps outbound bandwidth is occupied, resulting in an obvious attack effect.

Obtained Packets of a TLS Encrypted High-Rate Large Resource Request Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2018-12-13 16:22:54.224000	45.195.38.213	TLSv1	.73	968	50384	Application Data, Application Data
2018-12-13 16:22:54.522000	45.195.38.213	TLSv1	.73	968	50379	Application Data, Application Data
2018-12-13 16:22:54.523000	45.195.38.213	TLSv1	.73	968	50388	Application Data, Application Data
2018-12-13 16:22:54.524000	45.195.38.213	TLSv1	.73	968	50386	Application Data, Application Data
2018-12-13 16:22:54.525000	45.195.38.213	TLSv1	.73	968	50387	Application Data, Application Data
2018-12-13 16:22:54.545000	45.195.38.213	TLSv1	.73	968	50385	Application Data, Application Data
2018-12-13 16:22:54.551000	45.195.38.213	TLSv1	.73	968	50382	Application Data, Application Data
2018-12-13 16:22:54.588000	45.195.38.213	TLSv1	.73	968	50380	Application Data, Application Data
2018-12-13 16:22:54.594000	45.195.38.213	TLSv1	.73	976	50384	Application Data, Application Data
2018-12-13 16:22:54.602000	45.195.38.213	TLSv1	.73	968	50383	Application Data, Application Data
2018-12-13 16:22:54.609000	45.195.38.213	TLSv1	.73	952	50381	Application Data, Application Data
2018-12-13 16:22:54.762000	45.195.38.213	TLSv1	.73	848	50379	Application Data, Application Data
2018-12-13 16:22:54.764000	45.195.38.213	TLSv1	.73	848	50388	Application Data, Application Data
2018-12-13 16:22:54.790000	45.195.38.213	TLSv1	.73	968	50386	Application Data, Application Data
2018-12-13 16:22:54.845000	45.195.38.213	TLSv1	.73	880	50380	Application Data, Application Data
2018-12-13 16:22:55.100000	45.195.38.213	TLSv1	.73	968	50381	Application Data, Application Data
2018-12-13 16:22:55.179000	45.195.38.213	TLSv1	.73	976	50383	Application Data, Application Data
2018-12-13 16:22:55.329000	45.195.38.213	TLSv1	.73	944	50384	Application Data, Application Data
2018-12-13 16:22:55.583000	45.195.38.213	TLSv1	.73	1016	50388	Application Data, Application Data
2018-12-13 16:22:55.670000	45.195.38.213	TLSv1	.73	976	50383	Application Data, Application Data
2018-12-13 16:22:55.675000	45.195.38.213	TLSv1	.73	984	50381	Application Data, Application Data
2018-12-13 16:22:55.676000	45.195.38.213	TLSv1	.73	976	50379	Application Data, Application Data
2018-12-13 16:22:55.677000	45.195.38.213	TLSv1	.73	984	50386	Application Data, Application Data
2018-12-13 16:22:55.680000	45.195.38.213	TLSv1	.73	984	50382	Application Data, Application Data
2018-12-13 16:22:55.681000	45.195.38.213	TLSv1	.73	976	50385	Application Data, Application Data
2018-12-13 16:22:55.698000	45.195.38.213	TLSv1	.73	976	50387	Application Data, Application Data
2018-12-13 16:22:55.736000	45.195.38.213	TLSv1	.73	984	50384	Application Data, Application Data

» **High-rate requests for database query, consuming server performance**

When hackers attack e-banking apps, they frequently request URLs related to account security to make the server continuously perform database operations, consuming server performance.

- **High-rate POST requests**

  Password resetting through frequently sending POST requests is one of the common tactics for attacking e-banking apps.

Obtained Packets of an Attack Caused by Frequently Sending POST Requests for Password Resetting

Time	Source	Protocol	Destination	Length	Sport	Info
2018-12-17 01:33:36.961820	.11	HTTP	.32	149	33480	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.963076	.11	HTTP	.34	149	28035	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.963085	.11	HTTP	.33	149	34076	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.963222	.11	HTTP	.35	149	29773	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.964179	.11	HTTP	.32	149	29957	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.964625	.11	HTTP	.33	149	29307	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.964634	.11	HTTP	.34	149	28621	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.965175	.11	HTTP	.34	149	32962	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:36.973543	.11	HTTP	.35	149	33620	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.019258	.11	HTTP	.34	149	34548	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.023765	.11	HTTP	.35	149	34512	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.025013	.11	HTTP	.32	149	33480	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.041552	.11	HTTP	.32	149	29957	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.051675	.11	HTTP	.33	149	34076	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.075181	.11	HTTP	.32	149	33480	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.082104	.11	HTTP	.32	149	28639	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.138056	.11	HTTP	.34	149	34548	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.145106	.11	HTTP	.32	149	28639	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.152190	.11	HTTP	.33	149	29289	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.157212	.11	HTTP	.32	149	33480	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.208214	.11	HTTP	.32	149	28639	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.233460	.11	HTTP	.32	149	28639	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.246973	.11	HTTP	.34	149	29625	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)
2018-12-17 01:33:37.248593	.11	HTTP	.35	149	28315	POST /pweb/ResetLoginPwdComfirm.do HTTP/1.1 (application/x-www-form-urlencoded)

- **High-rate GET requests**
  Some financial enterprises do not pay much attention to the security of e-banking accounts and allow account login using the GET method. As a result, frequently sending GET requests for login becomes a major tactic to attack e-banking apps.

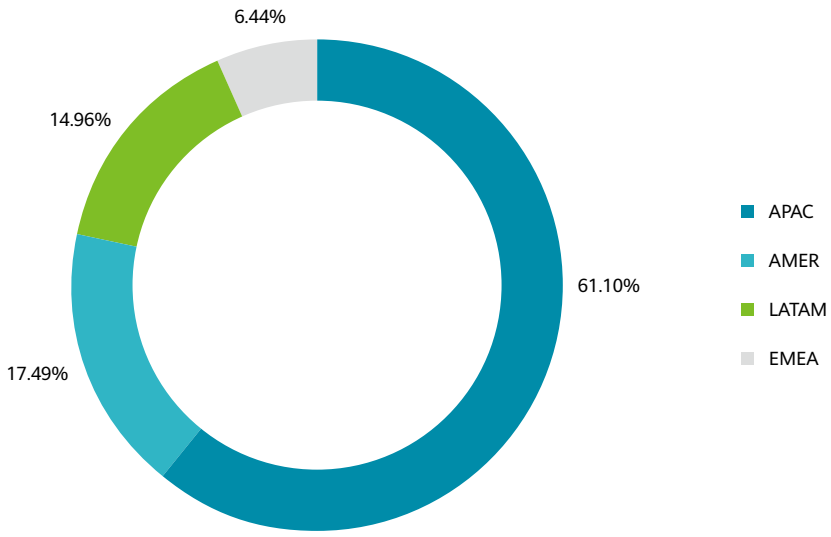
Obtained Packets of an Attack Caused by Frequently Sending GET Requests for Login

Time	Source	Protocol	Destination	Length	Sport	Info
2018-12-17 01:33:30.001108	.11	HTTP	.32	685	33480	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.004447	.11	HTTP	.33	751	33684	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.005054	.11	HTTP	.34	928	34548	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.008320	.11	HTTP	.35	770	32728	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.009969	.11	HTTP	.32	744	33480	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.011418	.11	HTTP	.33	897	29287	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.011543	.11	HTTP	.34	901	29621	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.015136	.11	HTTP	.35	861	29771	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.015154	.11	HTTP	.32	831	27441	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.022059	.11	HTTP	.33	850	33684	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.028752	.11	HTTP	.34	894	34548	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.029289	.11	HTTP	.33	748	26115	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.031411	.11	HTTP	.34	877	29621	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.031537	.11	HTTP	.35	829	32728	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.034163	.11	HTTP	.32	829	34290	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.034750	.11	HTTP	.35	867	29771	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.041065	.11	HTTP	.32	772	27433	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.041068	.11	HTTP	.33	774	34340	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.041190	.11	HTTP	.34	730	34548	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.041940	.11	HTTP	.33	897	29287	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.042144	.11	HTTP	.35	744	34512	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.043560	.11	HTTP	.32	785	34290	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.044548	.11	HTTP	.33	874	34076	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.045208	.11	HTTP	.34	715	34548	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1
2018-12-17 01:33:30.046636	.11	HTTP	.34	806	29621	GET /pweb/prelogin.do?LoginType=C&_locale=zh_CN&BankId=99999&VerifyIdNo=&VerifyIdType= HTTP/1.1

2. Attack source analysis

According to the source tracing of a financial attack incident, zombies are distributed in more than 80 countries and regions, with 74% of them distributed outside China (61.10% in APAC, 17.49% in AMER, 14.96% in LATAM, and 6.44% in EMEA).

Zombie Distribution



4.2.3 Analysis of Attacks Targeting the Finance Industry in 2020

In 2020, DDoS attacks on Chinese financial enterprises started in early February, impacting 11 banks, securities, and insurance enterprises, with portal websites and DNS servers being the main targets. Compared with the attack techniques in 2018, DDoS attacks in 2020 became more sophisticated. TCP reflection and TCP null connection attacks were newly adopted at the application layer.

1. Attack technique analysis

» **Combo TCP reflection attacks, congesting inbound network bandwidth**

In October 2020, a financial portal website suffered combo TCP reflection attacks, causing link bandwidth congestion. The source ports in TCP reflection attacks were mainly port 53, port 1900, and port 1723.

Obtained Packets of a Combo TCP Reflection Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-21 03:30:30.002370150	115.238.84.158	TCP	.49	68	1723	1723 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 SACK_PERM=1
2020-10-21 03:30:30.002370340	42.202.144.134	TCP	.49	68	53	53 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 SACK_PERM=1
2020-10-21 03:30:30.002371120	183.250.111.243	TCP	.49	68	1723	1723 → 80 [ACK] Seq=1 Ack=1 Win=2048 Len=0
2020-10-21 03:30:30.002371660	113.226.213.142	TCP	.49	70	1723	1723 → 80 [SYN, ACK] Seq=0 Ack=1 Win=6400 Len=0 MSS=1400
2020-10-21 03:30:30.002372520	111.53.55.76	TCP	.49	68	53	53 → 80 [ACK] Seq=1 Ack=1 Win=5840 Len=0
2020-10-21 03:30:30.002373100	125.211.147.151	TCP	.49	68	1900	1900 → 80 [ACK] Seq=1 Ack=1 Win=14400 Len=0
2020-10-21 03:30:30.002373700	183.233.128.138	TCP	.49	68	1723	1723 → 80 [ACK] Seq=1 Ack=1 Win=4096 Len=0
2020-10-21 03:30:30.002374540	120.197.12.100	TCP	.49	68	53	53 → 80 [ACK] Seq=1 Ack=1 Win=5640 Len=0
2020-10-21 03:30:30.002375070	116.226.240.34	TCP	.49	68	53	53 → 80 [ACK] Seq=1 Ack=1 Win=5808 Len=0
2020-10-21 03:30:30.002375990	121.28.92.98	TCP	.49	68	1723	1723 → 80 [ACK] Seq=1 Ack=1 Win=4800 Len=0
2020-10-21 03:30:30.002376390	117.25.158.235	TCP	.49	68	53	53 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 SACK_PERM=1
2020-10-21 03:30:30.002377190	61.147.228.208	TCP	.49	68	1900	1900 → 80 [ACK] Seq=1 Ack=1 Win=44456 Len=0
2020-10-21 03:30:30.002377900	120.193.93.25	TCP	.49	68	1723	1723 → 80 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2020-10-21 03:30:30.002378430	221.229.162.122	TCP	.49	68	1900	1900 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 SACK_PERM=1
2020-10-21 03:30:30.002379220	27.197.207.58	TCP	.49	68	1900	1900 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1440
2020-10-21 03:30:30.002379750	111.53.55.76	TCP	.49	68	53	[TCP Dup ACK 206#1] 53 → 80 [ACK] Seq=1 Ack=1 Win=5840 Len=0
2020-10-21 03:30:30.002380650	221.10.175.52	TCP	.49	68	1723	1723 → 80 [ACK] Seq=1 Ack=1 Win=4800 Len=0
2020-10-21 03:30:30.002381310	119.117.36.231	TCP	.49	68	1723	1723 → 80 [SYN, ACK] Seq=0 Ack=1 Win=28800 Len=0 MSS=1440
2020-10-21 03:30:30.002382240	60.170.245.148	TCP	.49	68	53	53 → 80 [ACK] Seq=1 Ack=1 Win=14600 Len=0
2020-10-21 03:30:30.002383210	111.61.51.41	TCP	.49	68	1723	1723 → 80 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2020-10-21 03:30:30.002383680	58.218.211.52	TCP	.49	68	53	53 → 80 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 SACK_PERM=1
2020-10-21 03:30:30.002383800	115.236.81.106	TCP	.49	68	1900	[TCP Dup ACK 166#1] 1900 → 80 [ACK] Seq=1 Ack=1 Win=14600 Len=0
2020-10-21 03:30:30.002384470	116.232.53.47	TCP	.49	68	1723	1723 → 80 [ACK] Seq=1 Ack=1 Win=5808 Len=0
2020-10-21 03:30:30.002385120	113.65.44.232	TCP	.49	68	1900	1900 → 80 [ACK] Seq=1 Ack=1 Win=28800 Len=0

When an attacker sends SYN packets to an open server on the Internet to trigger TCP reflection, if the server port is disabled, some operating systems reply with RST-ACK packets. If the server port is enabled, SYN-ACK, ACK, and RST packets are reflected with the sequence change of SYN packets.

» **High-rate large resource request attacks, consuming outbound network bandwidth**

In October 2020, a financial portal website was attacked by high-rate large resource request attacks, consuming the outbound bandwidth of the network where the server was located. The attacks involved fixed large resource request attacks and variable large resource request attacks.

Obtained Packets of a Fixed Large Resource Request Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-19 15:55:54.550000	112.1.196.75	HTTP	.49	954	13186	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:55:59.209000	165.84.180.47	HTTP	.49	905	39923	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:02.844000	124.251.48.3	HTTP	.49	867	2781	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:04.255000	183.254.101.109	HTTP	.49	914	6537	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:04.669000	111.22.244.10	HTTP	.49	867	29952	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:06.388000	120.194.180.94	HTTP	.49	666	7457	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:15.589000	112.1.196.75	HTTP	.49	954	13186	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:18.314000	183.254.101.109	HTTP	.49	914	6537	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:19.542000	165.84.180.47	HTTP	.49	905	28550	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:22.367000	124.251.48.3	HTTP	.49	867	2781	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:26.908000	120.194.180.94	HTTP	.49	666	7457	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:31.050000	111.22.244.10	HTTP	.49	867	29952	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:34.685000	183.254.101.109	HTTP	.49	914	6537	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:36.307000	112.1.196.75	HTTP	.49	954	13186	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:40.661000	165.84.180.47	HTTP	.49	905	27237	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:43.494000	124.251.48.3	HTTP	.49	867	2781	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:47.636000	120.194.180.94	HTTP	.49	666	7457	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:51.682000	111.22.244.10	HTTP	.49	867	29952	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:56:56.634000	112.1.196.75	HTTP	.49	954	13186	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:01.385000	165.84.180.47	HTTP	.49	905	31854	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:04.418000	124.251.48.3	HTTP	.49	867	2781	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:04.624000	183.254.101.109	HTTP	.49	914	6537	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:08.460000	120.194.180.94	HTTP	.49	666	7457	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:12.199000	111.22.244.10	HTTP	.49	867	29952	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:17.452000	112.1.196.75	HTTP	.49	954	13186	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:22.603000	165.84.180.47	HTTP	.49	905	52512	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1
2020-10-19 15:57:24.629000	124.251.48.3	HTTP	.49	867	2781	GET /static/ bank/images/pic/gd_newsicon2.png HTTP/1.1

As shown in the following figure, when variable large resource request attacks are launched, the attack URL contains a large number of images and a small number of CSS and JS files. The attack requests are the same as the access behaviors of normal browsers, indicating that the attacker launches the attacks by invoking scripts to access the portal website.

Obtained Packets of a Variable Large Resource Request Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-19 06:12:15.737000	60.6.209.219	HTTP	.234	380	60752	GET /site/uiFramework/huilan-jquery-ui/css/huilan-jquery-ui.css HTTP/1.1
2020-10-19 06:12:15.798000	60.6.209.219	HTTP	.234	421	60753	GET /site/uiFramework/huilan-jquery-ui/js/huilan-jquery-ui.js?self=true&skin=default HTTP/1.1
2020-10-19 06:12:15.807000	60.6.209.219	HTTP	.234	400	60755	GET /static/ bank/images/pic/serach_icon.png HTTP/1.1
2020-10-19 06:12:15.809000	60.6.209.219	HTTP	.234	388	60754	GET /site/uiFramework/js/counting/chanelCounting.js HTTP/1.1
2020-10-19 06:12:15.819000	60.6.209.219	HTTP	.234	398	60759	GET /static/ bank/images/pic/bank_logo.jpg HTTP/1.1
2020-10-19 06:12:15.870000	60.6.209.219	HTTP	.234	621	60752	GET /static/ bank/images/pic/subnav_a_hover.png HTTP/1.1
2020-10-19 06:12:15.881000	60.6.209.219	HTTP	.234	638	60753	GET /site/resource/cms/2019/03/img_pc_site/20190322143134217 69.jpg HTTP/1.1
2020-10-19 06:12:15.893000	60.6.209.219	HTTP	.234	638	60755	GET /site/resource/cms/2020/02/img_pc_site/20200224173820690 99.jpg HTTP/1.1
2020-10-19 06:12:15.894000	60.6.209.219	HTTP	.234	638	60757	GET /site/resource/cms/2020/08/img_pc_site/20200817140742608 02.jpg HTTP/1.1
2020-10-19 06:12:15.959000	60.6.209.219	HTTP	.234	638	60754	GET /site/resource/cms/2018/03/img_pc_site/20180329212417666 26.jpg HTTP/1.1
2020-10-19 06:12:15.964000	60.6.209.219	HTTP	.234	638	60756	GET /site/resource/cms/2018/09/img_pc_site/20180919084400279 73.jpg HTTP/1.1
2020-10-19 06:12:15.970000	60.6.209.219	HTTP	.234	638	60758	GET /site/resource/cms/2020/10/img_pc_site/20201015103408910 74.jpg HTTP/1.1
2020-10-19 06:12:15.971000	60.6.209.219	HTTP	.234	638	60761	GET /site/resource/cms/2020/09/img_pc_site/20200930154303754 26.jpg HTTP/1.1
2020-10-19 06:12:15.971000	60.6.209.219	HTTP	.234	638	60760	GET /site/resource/cms/2020/10/img_pc_site/20201013162937864 64.jpg HTTP/1.1
2020-10-19 06:12:15.973000	60.6.209.219	HTTP	.234	596	60759	GET /static/ bank/js/dataAcquisition.js HTTP/1.1
2020-10-19 06:12:15.974000	60.6.209.219	HTTP	.234	638	60752	GET /site/resource/cms/2020/08/img_pc_site/20200817144306697 47.jpg HTTP/1.1
2020-10-19 06:12:15.974000	60.6.209.219	HTTP	.234	638	60755	GET /site/resource/cms/2020/09/img_pc_site/20200929170323873 53.jpg HTTP/1.1
2020-10-19 06:12:15.975000	60.6.209.219	HTTP	.234	638	60753	GET /site/resource/cms/2020/10/img_pc_site/20201009114349730 05.JPG HTTP/1.1
2020-10-19 06:12:15.980000	60.6.209.219	HTTP	.234	638	60757	GET /site/resource/cms/2020/09/img_pc_site/20200917154657692 74.JPG HTTP/1.1
2020-10-19 06:12:15.980000	60.6.209.219	HTTP	.234	638	60754	GET /site/resource/cms/2020/09/img_pc_site/20200915083107592 15.png HTTP/1.1
2020-10-19 06:12:15.989000	60.6.209.219	HTTP	.234	616	60756	GET /static/ bank/images/pic/preButton.png HTTP/1.1
2020-10-19 06:12:15.990000	60.6.209.219	HTTP	.234	626	60761	GET /site/resource/cms/2017/01/20170116084705276 22.jpg HTTP/1.1
2020-10-19 06:12:15.994000	60.6.209.219	HTTP	.234	638	60759	GET /site/resource/cms/2018/04/img_pc_site/20180416175833644 51.jpg HTTP/1.1
2020-10-19 06:12:16.049000	60.6.209.219	HTTP	.234	626	60760	GET /site/resource/cms/2016/11/20161124195311462 12.jpg HTTP/1.1
2020-10-19 06:12:16.050000	60.6.209.219	HTTP	.234	617	60758	GET /static/ bank/images/pic/nextButton.png HTTP/1.1
2020-10-19 06:12:16.052000	60.6.209.219	HTTP	.234	626	60752	GET /site/resource/cms/2016/11/20161124195556313 48.jpg HTTP/1.1
2020-10-19 06:12:16.057000	60.6.209.219	HTTP	.234	626	60757	GET /site/resource/cms/2016/02/20160224162056562 99.jpg HTTP/1.1
2020-10-19 06:12:16.059000	60.6.209.219	HTTP	.234	626	60755	GET /site/resource/cms/2016/02/20160224161738406 59.jpg HTTP/1.1
2020-10-19 06:12:16.063000	60.6.209.219	HTTP	.234	618	60756	GET /static/ bank/images/pic/tzcpcx_icon.png HTTP/1.1
2020-10-19 06:12:16.066000	60.6.209.219	HTTP	.234	616	60761	GET /static/ bank/images/pic/cycx_icon.png HTTP/1.1
2020-10-19 06:12:16.121000	60.6.209.219	HTTP	.234	626	60753	GET /site/resource/cms/2016/02/20160224161509734 17.jpg HTTP/1.1
2020-10-19 06:12:16.122000	60.6.209.219	HTTP	.234	626	60754	GET /site/resource/cms/2016/02/20160224162002897 65.jpg HTTP/1.1
2020-10-19 06:12:16.134000	60.6.209.219	HTTP	.234	616	60759	GET /static/ bank/images/pic/yybl_icon.png HTTP/1.1
2020-10-19 06:12:16.135000	60.6.209.219	HTTP	.234	616	60760	GET /static/ bank/images/pic/ljkt_icon.png HTTP/1.1
2020-10-19 06:12:16.135000	60.6.209.219	HTTP	.234	616	60758	GET /static/ bank/images/pic/shjf_icon.png HTTP/1.1

» **TCP null connection attacks, consuming Layer 4 session resources**

In October 2020, a financial portal website was attacked by TCP null connection attacks. The attacker established a TCP connection with the target server through zombie hosts and immediately disconnected the connection, consuming TCP session resources of the target server and network devices.

Obtained Packets of a TCP Null Connection Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-19 15:55:51.308000	125.212.249.132	TCP	.49	66	53864	53864 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2020-10-19 15:55:51.310000	125.212.249.132	TCP	.49	54	53864	53864 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.311000	125.212.249.132	TCP	.49	54	53864	53864 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.311000	125.212.249.132	TCP	.49	66	54317	54317 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2020-10-19 15:55:51.312000	125.212.249.132	TCP	.49	54	53864	53864 → 443 [ACK] Seq=2 Ack=2 Win=65536 Len=0
2020-10-19 15:55:51.313000	125.212.249.132	TCP	.49	54	54317	54317 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.314000	125.212.249.132	TCP	.49	54	54317	54317 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.314000	125.212.249.132	TCP	.49	54	54317	54317 → 443 [ACK] Seq=2 Ack=2 Win=65536 Len=0
2020-10-19 15:55:51.315000	106.120.215.201	TCP	.49	74	45642	45642 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1100106427 TSecr=0 WS=128
2020-10-19 15:55:51.316000	103.22.181.94	TCP	.49	66	60452	60452 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1340 WS=256 SACK_PERM=1
2020-10-19 15:55:51.316000	125.212.249.132	TCP	.49	66	59739	59739 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=0 SLE=0 SRE=1
2020-10-19 15:55:51.317000	103.22.181.94	TCP	.49	54	60452	60452 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.318000	103.22.181.94	TCP	.49	54	60452	60452 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:51.318000	125.212.249.132	TCP	.49	66	55269	55269 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2020-10-19 15:55:52.925000	103.22.181.94	TCP	.49	54	60452	60452 → 443 [ACK] Seq=2 Ack=2 Win=65536 Len=0
2020-10-19 15:55:52.926000	125.212.249.132	TCP	.49	66	55466	55466 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2020-10-19 15:55:52.927000	125.212.249.132	TCP	.49	54	55269	55269 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:52.927000	125.212.249.132	TCP	.49	54	55269	55269 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:52.928000	125.212.249.132	TCP	.49	54	55466	55466 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:52.928000	125.212.249.132	TCP	.49	54	55466	55466 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0
2020-10-19 15:55:52.929000	125.212.249.132	TCP	.49	54	55269	55269 → 443 [ACK] Seq=2 Ack=2 Win=65536 Len=0
2020-10-19 15:55:52.930000	103.22.181.94	TCP	.49	66	60515	60515 → 443 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=1340 WS=256 SACK_PERM=1
2020-10-19 15:55:52.930000	125.212.249.132	TCP	.49	54	55466	55466 → 443 [ACK] Seq=2 Ack=2 Win=65536 Len=0
2020-10-19 15:55:52.931000	103.22.181.94	TCP	.49	54	60515	60515 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0

» **Incomplete TLS session attacks, consuming load balancing resources**

Most financial enterprises use load balancing devices to implement TLS encryption and decryption. When attackers launch incomplete TLS session attacks, TLS session resources and processing performance of load balancing devices are directly consumed. There are two patterns of incomplete TLS session attacks. One is that a TCP connection is disconnected after only Client Hello packets are sent. The other is that the connection is disconnected after the TLS handshake is completed. In October 2020, both attack patterns were used during the attacks targeting portal websites of financial enterprises.

Obtained Packets of an Incomplete TLS Session Attack: No Application Data Transmission after TLS Handshake

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-20 03:40:25.141000	117.136.31.144	TCP	.49	66	9464	9464 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1360 SACK_PERM=1 WS=256
2020-10-20 03:40:25.153000	117.136.31.144	TCP	.49	54	9464	9464 → 443 [ACK] Seq=1 Ack=1 Win=81664 Len=0
2020-10-20 03:40:25.292000	117.136.31.144	TLSv1.2	.49	253	9464	Client Hello
2020-10-20 03:40:25.295000	117.136.31.144	TCP	.49	54	9464	9464 → 443 [ACK] Seq=200 Ack=1361 Win=84480 Len=0
2020-10-20 03:40:25.295000	117.136.31.144	TCP	.49	54	9464	9464 → 443 [ACK] Seq=200 Ack=3318 Win=88320 Len=0
2020-10-20 03:40:25.310000	117.136.31.144	TLSv1.2	.49	180	9464	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
2020-10-20 03:40:25.488000	117.136.31.144	TCP	.49	54	9464	9464 → 443 [ACK] Seq=326 Ack=3592 Win=91136 Len=0
2020-10-20 03:40:27.544000	117.136.31.144	TCP	.49	54	9464	9464 → 443 [RST, ACK] Seq=326 Ack=3592 Win=91136 Len=0
2020-10-20 03:40:25.140000	117.136.31.144	TCP	.49	66	9465	9465 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1360 SACK_PERM=1 WS=256
2020-10-20 03:40:25.152000	117.136.31.144	TCP	.49	54	9465	9465 → 443 [ACK] Seq=1 Ack=1 Win=81664 Len=0
2020-10-20 03:40:25.152000	117.136.31.144	TLSv1.2	.49	253	9465	Client Hello
2020-10-20 03:40:25.293000	117.136.31.144	TCP	.49	54	9465	9465 → 443 [ACK] Seq=200 Ack=2721 Win=87040 Len=0
2020-10-20 03:40:25.293000	117.136.31.144	TCP	.49	54	9465	9465 → 443 [ACK] Seq=200 Ack=3318 Win=89856 Len=0
2020-10-20 03:40:25.294000	117.136.31.144	TLSv1.2	.49	180	9465	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
2020-10-20 03:40:25.324000	117.136.31.144	TCP	.49	54	9465	9465 → 443 [ACK] Seq=326 Ack=3592 Win=92672 Len=0
2020-10-20 03:40:27.543000	117.136.31.144	TCP	.49	54	9465	9465 → 443 [RST, ACK] Seq=326 Ack=3592 Win=92672 Len=0
2020-10-20 03:40:25.140000	117.136.31.144	TCP	.49	66	9466	9466 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1360 SACK_PERM=1 WS=256
2020-10-20 03:40:25.153000	117.136.31.144	TCP	.49	54	9466	9466 → 443 [ACK] Seq=1 Ack=1 Win=81664 Len=0
2020-10-20 03:40:25.154000	117.136.31.144	TLSv1.2	.49	253	9466	Client Hello
2020-10-20 03:40:25.294000	117.136.31.144	TCP	.49	54	9466	9466 → 443 [ACK] Seq=200 Ack=3318 Win=88320 Len=0
2020-10-20 03:40:25.296000	117.136.31.144	TLSv1.2	.49	180	9466	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
2020-10-20 03:40:25.324000	117.136.31.144	TCP	.49	54	9466	9466 → 443 [ACK] Seq=326 Ack=3592 Win=91136 Len=0
2020-10-20 03:40:27.544000	117.136.31.144	TCP	.49	54	9466	9466 → 443 [RST, ACK] Seq=326 Ack=3592 Win=91136 Len=0

Obtained Packets of an Incomplete TLS Session Attack: Only Client Hello Packets Are Sent

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-20 03:40:39.370000	221.181.214.168	TCP	.49	66	60472	60472 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM=1
2020-10-20 03:40:39.495000	221.181.214.168	TCP	.49	54	60472	60472 → 443 [ACK] Seq=1 Ack=1 Win=372296 Len=0
2020-10-20 03:40:39.496000	221.181.214.168	TLSv1	.49	131	60472	Client Hello
2020-10-20 03:40:39.496000	221.181.214.168	TCP	.49	54	60472	60472 → 443 [ACK] Seq=78 Ack=9 Win=372288 Len=0
2020-10-20 03:40:39.497000	221.181.214.168	TCP	.49	54	60472	60472 → 443 [FIN, ACK] Seq=78 Ack=9 Win=372288 Len=0

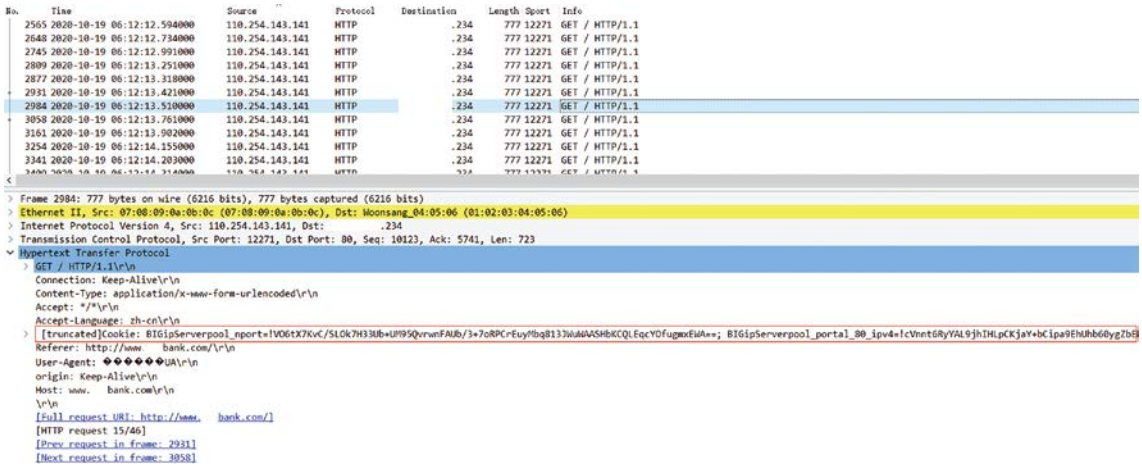
» **Attacks using high-rate requests for the root directory, consuming load balancing resources**

In October 2020, application-layer attacks targeting financial portal websites were launched through root directory attacks. Based on the adopted methods, root directory attacks can be classified into GET requests to the root directory and POST requests to the root directory.

- **Attacks using high-rate GET requests to the root directory**

  As shown in the following figure, after a TCP session is established between the attacking device and the target server, the attacker repeatedly requests the root directory. The attack packets carry the cookies of WAFs commonly used by the financial enterprise to bypass WAF filtering.

Obtained Packets of an Attack Repeatedly Requesting the Root Directory During One Session



As shown in the following figure, after a TCP session is established between the attacking device and the target server, the attacker sends a request to the root directory and immediately sends an RST packet to forcibly release its session resources. However, the server needs to spend more time clearing the response cache, resulting in excessive resource consumption.

Obtained Packets of an Attack Requesting the Root Directory Only Once During One Session

Time	Source	Protocol	Destination	Length	Sport	Info
2020-10-19 06:12:20.324000	221.220.175.129	TCP	.234	66	9804	9804 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:20.325000	221.220.175.129	TCP	.234	54	9804	9804 → 80 [ACK] Seq=1 Ack=1 Win=16776960 Len=0
2020-10-19 06:12:20.325000	221.220.175.129	HTTP	.234	199	9804	GET / HTTP/1.1
2020-10-19 06:12:20.327000	221.220.175.129	TCP	.234	54	9804	9804 → 80 [ACK] Seq=146 Ack=2629 Win=16776960 Len=0
2020-10-19 06:12:20.327000	221.220.175.129	TCP	.234	54	9804	9804 → 80 [FIN, ACK] Seq=146 Ack=2629 Win=16776960 Len=0
2020-10-19 06:12:20.328000	221.220.175.129	TCP	.234	54	9804	9804 → 80 [RST, ACK] Seq=147 Ack=2629 Win=0 Len=0
2020-10-19 06:12:20.358000	221.220.175.129	TCP	.234	66	9805	9805 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:20.586000	221.220.175.129	TCP	.234	66	9805	[TCP Retransmission] 9805 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:20.816000	221.220.175.129	TCP	.234	66	9805	[TCP Retransmission] 9805 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:21.448000	221.220.175.129	TCP	.234	66	9805	[TCP Retransmission] 9805 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:21.871000	221.220.175.129	TCP	.234	66	9805	[TCP Retransmission] 9805 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:21.900000	221.220.175.129	TCP	.234	66	9807	9807 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2020-10-19 06:12:22.084000	103.3.97.161	TCP	.234	66	19969	19969 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1360 WS=4 SACK_PERM=1
2020-10-19 06:12:22.093000	103.3.97.161	TCP	.234	54	19969	19969 → 80 [ACK] Seq=1 Ack=1 Win=261120 Len=0
2020-10-19 06:12:22.095000	103.3.97.161	HTTP	.234	321	19969	GET / HTTP/1.1
2020-10-19 06:12:22.128000	103.3.97.161	TCP	.234	54	19969	19969 → 80 [ACK] Seq=268 Ack=1586 Win=261120 Len=0
2020-10-19 06:12:22.128000	103.3.97.161	TCP	.234	54	19969	19969 → 80 [RST, ACK] Seq=268 Ack=2629 Win=0 Len=0

- **Attacks using high-rate POST requests to the root directory**
  In addition to causing an unbalanced load balancing effect, the attacks make the server respond with 40X HTTP status codes to unsupported requests, consuming server performance and bandwidth.

Obtained Packets of an Attack Using POST Requests for the Root Directory

No.	Time	Source	Protocol	Destination	Length	Sport	Info
157	2020-10-19 06:12:08.324000	1.204.206.2	HTTP	.234	65	24247	POST / HTTP/1.1 (application/x-www-form-urlencoded)
224	2020-10-19 06:12:08.557000	1.204.206.2	TCP	.234	540	24247	24247 → 80 [PSH, ACK] Seq=995 Ack=821 Win=65340 Len=486 [TCP segment of a reassembled PDU]
227	2020-10-19 06:12:08.558000	1.204.206.2	HTTP	.234	65	24247	POST / HTTP/1.1 (application/x-www-form-urlencoded)
311	2020-10-19 06:12:08.770000	1.204.206.2	TCP	.234	540	24247	24247 → 80 [PSH, ACK] Seq=1492 Ack=1231 Win=64930 Len=486 [TCP segment of a reassembled PDU]
312	2020-10-19 06:12:08.771000	1.204.206.2	HTTP	.234	65	24247	POST / HTTP/1.1 (application/x-www-form-urlencoded)
373	2020-10-19 06:12:08.852000	1.204.206.2	TCP	.234	540	24247	24247 → 80 [PSH, ACK] Seq=1989 Ack=1641 Win=64520 Len=486 [TCP segment of a reassembled PDU]
379	2020-10-19 06:12:08.855000	1.204.206.2	HTTP	.234	65	24247	POST / HTTP/1.1 (application/x-www-form-urlencoded)
435	2020-10-19 06:12:09.068000	1.204.206.2	TCP	.234	540	24247	24247 → 80 [PSH, ACK] Seq=2486 Ack=2051 Win=64110 Len=486 [TCP segment of a reassembled PDU]
436	2020-10-19 06:12:09.069000	1.204.206.2	HTTP	.234	65	24247	POST / HTTP/1.1 (application/x-www-form-urlencoded)
504	2020-10-19 06:12:09.174000	1.204.206.2	TCP	.234	540	24247	24247 → 80 [PSH, ACK] Seq=2983 Ack=2461 Win=65340 Len=486 [TCP segment of a reassembled PDU]

```
> Frame 157: 65 bytes on wire (520 bits), 65 bytes captured (520 bits)
> Ethernet II, Src: 07:08:09:0a:0b:0c (07:08:09:0a:0b:0c), Dst: Woonsang_04:05:06 (01:02:03:04:05:06)
> Internet Protocol Version 4, Src: 1.204.206.2, Dst:        .234
> Transmission Control Protocol, Src Port: 24247, Dst Port: 80, Seq: 984, Ack: 411, Len: 11
> [2 Reassembled TCP Segments (497 bytes): #156(486), #157(11)]
v Hypertext Transfer Protocol
  > POST / HTTP/1.1\r\n
    Connection: Keep-Alive\r\n
    Content-Type: application/x-www-form-urlencoded\r\n
    Accept: */*\r\n
    Accept-Language: zh-cn\r\n
  > [truncated]Cookie: BIGipServerpool_nport=!Gr31gC1EauQasa/H33Ub+UM95Qvrwq8XmPKqsLp4V+dZ4ZLY3ksTdTLbGM8Hzgqq7EvyA/pjcnTBTw==; BIGipServerpool_waf_nport=!LJwUODlDdxwsZgrH33Ub+UM95Qvr
    Referer: http://www.    bank.com/\r\n
    User-Agent: 2?������?UA\r\n
    origin: Keep-Alive\r\n
  > Content-Length: 11\r\n
    Host: www.    bank.com\r\n
    \r\n
    [Full request URI: http://www.    bank.com/]
    [HTTP request 2/92]
    [Prev request in frame: 93]
    [Next request in frame: 227]
    File Data: 11 bytes
v HTML Form URL Encoded: application/x-www-form-urlencoded
  v Form item: "post~?y?Y" = ""
      Key: post~?y?Y
      Value:
```

» **DNS NXDOMAIN flood attacks, consuming DNS server performance**
If a non-existent domain name is requested, the DNS server returns an NXDOMAIN error message, which is also called the "No such name" message, indicating that the domain name does not exist. This may cause more server resources to be consumed. Therefore, DNS query flood attacks that request non-existent domain names are also called DNS NXDOMAIN flood attacks.

In November 2020, the DNS server of a financial enterprise suffered DNS NXDOMAIN flood attacks.

Obtained Packets of a DNS NXDOMAIN Flood Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2020-11-16 06:04:04.502240970	.107	DNS	173.194.93.3	153	53	Standard query response 0xc342 AAAA dns3. bank.com SOA dns3. bank.com OPT
2020-11-16 06:04:04.504651520	114.67.160.244	DNS	.107	95	51461	Standard query 0x4d3e A usbl. bank.com OPT
2020-11-16 06:04:04.504798190	114.67.160.244	DNS	.107	95	50250	Standard query 0xdfe4 A s521. bank.com OPT
2020-11-16 06:04:04.505109490	.107	DNS	114.67.160.244	147	53	Standard query response 0x4d3e No such name A usbl. bank.com SOA dns3. bank.com OPT
2020-11-16 06:04:04.505678770	74.125.41.6	DNS	.107	106	49119	Standard query 0x9041 AAAA dns4. bank.com OPT
2020-11-16 06:04:04.505911480	.107	DNS	114.67.160.244	147	53	Standard query response 0xdfe4 No such name A s521. bank.com SOA dns3. bank.com OPT
2020-11-16 06:04:04.505937110	114.67.160.244	DNS	.107	95	2480	Standard query 0x87a5 A nt58. bank.com OPT
2020-11-16 06:04:04.505997860	114.67.160.244	DNS	.107	92	31968	Standard query 0x8419 A rz10. 95.cn OPT
2020-11-16 06:04:04.506012500	114.67.160.244	DNS	.107	95	33437	Standard query 0x8ed9 A 0bsq. bank.com OPT
2020-11-16 06:04:04.506108170	114.67.160.244	DNS	.107	98	55759	Standard query 0x5a5d A jobbery. bank.com OPT
2020-11-16 06:04:04.506400560	.107	DNS	114.67.160.244	147	53	Standard query response 0x87a5 No such name A nt58. bank.com SOA dns3. bank.com OPT
2020-11-16 06:04:04.506508070	114.67.160.244	DNS	.107	92	9417	Standard query 0x0714 A cmrn. 95.cn OPT
2020-11-16 06:04:04.506647600	114.67.160.244	DNS	.107	101	22134	Standard query 0xeb33 A mail-batch. bank.com OPT
2020-11-16 06:04:04.506648590	114.67.160.244	DNS	.107	97	38917	Standard query 0x3df1 A trough. bank.com OPT
2020-11-16 06:04:04.506699240	.107	DNS	114.67.160.244	150	53	Standard query response 0x5a5d No such name A jobbery. bank.com SOA dns3. bank.com OPT

2. Attack source analysis

In 2020, among all application-layer DDoS attacks on Chinese financial enterprises, more than 90% of zombies came from China.

4.2.4 Analysis of Attacks Targeting the Finance Industry in 2021

In 2021, DDoS attacks on Chinese financial enterprises started in January, impacting 24 banks, securities, and insurance enterprises with portal websites and financial services systems being the main targets. The attack complexity was further increased, and a variant of network-layer CC attacks occurred for the first time.

1. Attack technique analysis

» **DNS reflection attacks, congesting inbound network bandwidth**

In February 2021, a financial portal website suffered DNS reflection attacks.

As shown in the following figure, the DNS request type is ANY and a large number of fragmented packets are contained. The attacker sets the DNS request type to ANY to trigger the DNS server to respond with all records (such as A, AAAA, MX, NS, SOA, TXT, and RRSIG) for an intensified attack effect. In addition, the attacker sets EDNS0 extension for DNS request packets to break through the 512-byte limit by UDP for DNS message transmission, inducing fragment attacks.

Obtained Packets of a DNS Reflection Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2021-02-03 13:52:44.723000	85.175.72.155	IPv4	.47	926		Fragmented IP protocol (proto=UDP 17, off=2912, ID=8b8c)
2021-02-03 13:52:44.738000	89.106.106.27	IPv4	.47	1328		Fragmented IP protocol (proto=UDP 17, off=1480, ID=1c80)
2021-02-03 13:52:44.738000	82.149.203.82	IPv4	.47	1328		Fragmented IP protocol (proto=UDP 17, off=1480, ID=831a)
2021-02-03 13:52:44.739000	62.220.59.83	DNS	.47	1328	53	Standard query response 0xc406 ANY vtk.be MX 1 aspmx.l.google.com RRSIG RRSIG RRSIG RRSIG[Unreassembled Packet]
2021-02-03 13:52:44.739000	203.176.135.180	IPv4	.47	1025		Fragmented IP protocol (proto=UDP 17, off=2912, ID=5de8)
2021-02-03 13:52:44.739000	103.248.41.58	IPv4	.47	977		Fragmented IP protocol (proto=UDP 17, off=2960, ID=d90b)
2021-02-03 13:52:44.754000	103.210.64.29	DNS	.47	1328	53	Standard query response 0x8ffa ANY vtk.be RRSIG RRSIG RRSIG[Unreassembled Packet]
2021-02-03 13:52:44.755000	78.11.38.242	IPv4	.47	873		Fragmented IP protocol (proto=UDP 17, off=2960, ID=9c36)
2021-02-03 13:52:44.755000	77.51.178.156	IPv4	.47	1328		Fragmented IP protocol (proto=UDP 17, off=1376, ID=42fb)
2021-02-03 13:52:44.756000	186.97.152.194	DNS	.47	1328	53	Standard query response 0x8ffa ANY vtk.be RRSIG RRSIG RRSIG[Unreassembled Packet]
2021-02-03 13:52:44.756000	92.51.90.245	IPv4	.47	1328		Fragmented IP protocol (proto=UDP 17, off=1480, ID=13a5)
2021-02-03 13:52:44.757000	92.86.180.35	IPv4	.47	1032		Fragmented IP protocol (proto=UDP 17, off=2960, ID=05a8)
2021-02-03 13:52:44.779000	154.117.166.182	IPv4	.47	914		Fragmented IP protocol (proto=UDP 17, off=2960, ID=0e95)
2021-02-03 13:52:44.780000	200.54.42.3	IPv4	.47	998		Fragmented IP protocol (proto=UDP 17, off=2960, ID=df86)
2021-02-03 13:52:44.781000	103.248.41.58	IPv4	.47	956		Fragmented IP protocol (proto=UDP 17, off=2960, ID=d90c)
2021-02-03 13:52:44.781000	213.155.195.48	IPv4	.47	984		Fragmented IP protocol (proto=UDP 17, off=2752, ID=51b6)
2021-02-03 13:52:44.782000	82.204.140.154	IPv4	.47	1328		Fragmented IP protocol (proto=UDP 17, off=1480, ID=d540)
2021-02-03 13:52:44.803000	202.91.43.185	DNS	.47	1328	53	Standard query response 0xd958 ANY vtk.be DNSKEY DNSKEY DNSKEY[Unreassembled Packet]

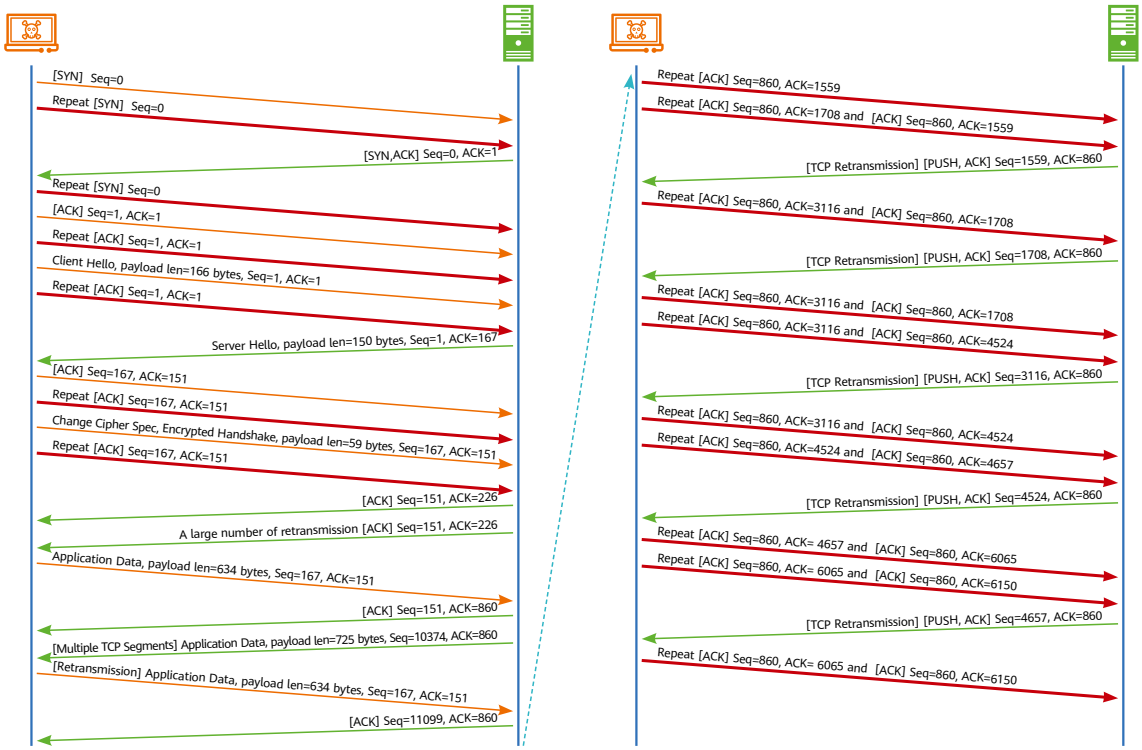
» **Variant network-layer CC attacks, occupying inbound and outbound network bandwidth**

In April 2021, a new type of network-layer CC attacks emerged in attacks on the e-banking system of a financial enterprise. The network-layer CC attacks caused an obvious attack effect and stronger defense evasion capabilities.

The following figure shows the interaction between an attacking device and the server. There is a complete TLS interaction process during the session. However, the same packet is repeatedly sent at a

high rate, which does not comply with the TCP interaction logic. This kind of attack evolves from network-layer CC attacks launched by Mirai botnets. That is, when the attacking device exchanges complete HTTPS data with the server through sockets, the SYN and ACK packets are replayed at a high rate through the raw socket, forming flood attacks on servers.

Variant Network-Layer CC Attack: Interaction Between an Attacking Device and the Server



According to the obtained attack packets during high-rate SYN packet replay, even if the server has replied with a SYN-ACK packet, the attacking device still replays the SYN packet in the same session.

Variant Network-Layer CC Attack: Obtained Packets of an Attack During High-Rate SYN Packet Replay

	Time	Source	Protocol	Destination	Length	Sport	Info
1	2021-04-09 05:30:42.207388700	171.105.175.191	TCP	.150	74	9323	9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM=1
2	2021-04-09 05:30:42.207390340	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
3	2021-04-09 05:30:42.207392770	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
4	2021-04-09 05:30:42.207392840	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
5	2021-04-09 05:30:42.207394070	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
6	2021-04-09 05:30:42.207394410	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
7	2021-04-09 05:30:42.207408190	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
8	2021-04-09 05:30:42.207410030	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
9	2021-04-09 05:30:42.207410880	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
10	2021-04-09 05:30:42.207412260	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
11	2021-04-09 05:30:42.207412400	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
12	2021-04-09 05:30:42.207413580	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
13	2021-04-09 05:30:42.207414230	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
14	2021-04-09 05:30:42.207414310	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
15	2021-04-09 05:30:42.207415280	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
16	2021-04-09 05:30:42.207416540	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
17	2021-04-09 05:30:42.207416610	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
18	2021-04-09 05:30:42.207417130	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
51	2021-04-09 05:30:42.207721610	.150	TCP	171.105.175.191	68	443	443 → 9323 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
52	2021-04-09 05:30:42.207865250	.150	TCP	171.105.175.191	68	443	[TCP Out-Of-Order] 443 → 9323 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
53	2021-04-09 05:30:42.207865590	.150	TCP	171.105.175.191	68	443	[TCP Out-Of-Order] 443 → 9323 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
54	2021-04-09 05:30:42.208003420	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
55	2021-04-09 05:30:42.208005060	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
56	2021-04-09 05:30:42.208006700	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
57	2021-04-09 05:30:42.208006780	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
58	2021-04-09 05:30:42.208007100	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
59	2021-04-09 05:30:42.208035210	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM
60	2021-04-09 05:30:42.208036950	171.105.175.191	TCP	.150	74	9323	[TCP Out-Of-Order] 9323 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1412 WS=256 SACK_PERM

This variant of network-layer CC attacks further increases the threats of DDoS attacks. The high-rate ACK packet replay overloads the inbound bandwidth of the network where the attack target resides. In addition, the high-rate ACK packet replay triggers the fast retransmission mechanism of the target server, consuming the outbound bandwidth of the network where the attack target resides.

Variant Network-Layer CC Attack: Obtained Packets of an Attack During High-Rate ACK Packet Replay

Time	Source	Protocol	Destination	Length	Sport	Info
2021-04-09 05:30:42.470663400	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#117] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470664390	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#118] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470664790	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#119] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470665380	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#120] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470666360	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#121] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470667010	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#122] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470668270	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#123] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470668400	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#124] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470669050	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#125] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470669770	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#126] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470670230	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#127] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470670820	171.105.175.191	TCP	.150	68	9323	[TCP Dup ACK 1179#128] 9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470672260	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470672330	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470673120	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470673520	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470674820	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470674960	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470675740	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470788100	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470788220	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470788320	.150	TCP	171.105.175.191	1470	443	[TCP Retransmission] 443 → 9323 [PSH, ACK] Seq=1559 Ack=860 Win=65535 Len=1408
2021-04-09 05:30:42.470788870	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0
2021-04-09 05:30:42.470789790	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1708 Win=16589568 Len=0
2021-04-09 05:30:42.470790320	171.105.175.191	TCP	.150	68	9323	9323 → 443 [ACK] Seq=860 Ack=1559 Win=16627712 Len=0

» **Encrypted attacks, bypassing the CDN for defense evasion**

Generally, financial enterprises use CDNs to accelerate HTTP and HTTPS access. However, most CDNs do not provide DDoS protection capabilities. When an attacker intentionally requests resources that do not exist, the CDN has to redirect the requests to the server. As a result, application-layer attacks bypass the CDN and penetrated to the target server.

In February 2021, application-layer encrypted attacks on a financial service app bypassed the CDN and penetrated to the server.

Obtained Packets of an Attack Bypassing the CDN

Time	Source	Protocol	Destination	Length	Sport	Info
2021-02-24 00:22:00.917169832	222.	TLSv1.2	219.	1058	24823	Application Data
2021-02-24 00:22:16.809983010	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:19.790402220	222.	TLSv1.2	219.	1060	24823	Application Data
2021-02-24 00:22:19.810447105	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:20.269306475	222.	TLSv1.2	219.	1056	24823	Application Data
2021-02-24 00:22:20.291789787	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:20.331682226	222.	TLSv1.2	219.	1010	24823	Application Data
2021-02-24 00:22:20.355175038	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:20.412458194	222.	TLSv1.2	219.	1057	24823	Application Data
2021-02-24 00:22:20.432568924	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:20.897455315	222.	TLSv1.2	219.	1175	24823	Application Data
2021-02-24 00:22:20.919486929	219.	TLSv1.2	222.	645	443	Application Data
2021-02-24 00:22:20.989337277	222.	TLSv1.2	219.	1056	24823	Application Data
2021-02-24 00:22:21.010964936	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:21.160078608	222.	TLSv1.2	219.	1059	24823	Application Data
2021-02-24 00:22:21.179737089	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:21.212048384	222.	TLSv1.2	219.	1008	24823	Application Data
2021-02-24 00:22:21.234362616	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:21.486511504	222.	TLSv1.2	219.	1056	24823	Application Data
2021-02-24 00:22:21.507581490	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:21.705879820	222.	TLSv1.2	219.	1054	24823	Application Data
2021-02-24 00:22:21.725855838	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:22.008718155	222.	TLSv1.2	219.	1080	24823	Application Data
2021-02-24 00:22:22.030206898	219.	TLSv1.2	222.	645	443	Application Data
2021-02-24 00:22:22.059327528	222.	TLSv1.2	219.	1052	24823	Application Data
2021-02-24 00:22:22.081728340	219.	TLSv1.2	222.	646	443	Application Data
2021-02-24 00:22:22.118426213	222.	TLSv1.2	219.	1181	24823	Application Data

2. Attack source analysis

In 2021, among all application-layer DDoS attacks on Chinese financial enterprises, more than 90% of zombies came from China.

4.2.5 Analysis of Attacks Targeting the Finance Industry in 2022

It is known by all that large-scale cyber attacks against financial enterprises in 2022 were mainly triggered by geopolitical conflicts, and Chinese financial enterprises also bore the brunt. In addition, with global hackers taking sides, the complexity of DDoS attacks further increased.

In 2022, a total of 4012 DDoS attacks targeting the finance industry occurred in the Chinese mainland, impacting 102 banks, securities, and insurance enterprises with portal websites and DNS servers being the main targets. DDoS attacks with the highest attack intensity occurred in November that year, mainly involving combo SYN flood attacks, RST flood attacks, HTTP & HTTPS application-layer attacks, and TLS attacks. The attacks, targeting portal websites, used eight attack vectors and the peak attack traffic reached 196 Gbps.

In 2022, the most extensive DDoS attacks occurred in February. The attack target was a financial enterprise outside China. During the attacks, a large number of geographically scattered zombies were used. These low-rate attacks were launched from a single source IP address and with stronger defense evasion capabilities.

1. Attack technique analysis

» **SYN carpet-bombing attacks, congesting inbound network bandwidth and consuming and Layer 4 session resources**

Since the fourth quarter of 2021, a Chinese financial enterprise has been frequently attacked by SYN carpet-bombing attacks until the end of 2022. When SYN carpet-bombing attacks occur, network links are congested, and each attack lasts for no more than 5 minutes.

Obtained Packets of an SYN Carpet-Bombing Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2021-10-28 07:17:09.831000	106.14.80.233	TCP	.243	74	40530	40530 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707749 TSecr=0 WS=128
2021-10-28 07:17:09.832000	106.14.80.233	TCP	.243	74	40592	40592 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707753 TSecr=0 WS=128
2021-10-28 07:17:09.833000	106.14.80.233	TCP	.243	74	40686	40686 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707761 TSecr=0 WS=128
2021-10-28 07:17:09.833000	121.37.198.224	TCP	.4	74	50401	50401 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1463817010 TSecr=0 WS=128
2021-10-28 07:17:09.842000	106.14.80.233	TCP	.77	74	60427	60427 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707762 TSecr=0 WS=128
2021-10-28 07:17:09.847000	106.14.80.233	TCP	.132	74	62509	62509 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707745 TSecr=0 WS=128
2021-10-28 07:17:09.848000	106.14.80.233	TCP	.132	74	62689	62689 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707756 TSecr=0 WS=128
2021-10-28 07:17:09.848000	106.14.80.233	TCP	.132	74	62717	62717 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707759 TSecr=0 WS=128
2021-10-28 07:17:09.857000	106.14.80.233	TCP	.242	74	50832	50832 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707752 TSecr=0 WS=128
2021-10-28 07:17:09.858000	106.14.80.233	TCP	.242	74	50890	50890 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707756 TSecr=0 WS=128
2021-10-28 07:17:09.858000	106.14.80.233	TCP	.182	74	36593	36593 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707756 TSecr=0 WS=128
2021-10-28 07:17:09.858000	106.14.80.233	TCP	.242	74	50970	50970 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707764 TSecr=0 WS=128
2021-10-28 07:17:09.858000	106.14.80.233	TCP	.182	74	36697	36697 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707772 TSecr=0 WS=128
2021-10-28 07:17:09.867000	121.37.198.224	TCP	.219	74	59525	59525 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1463817038 TSecr=0 WS=128
2021-10-28 07:17:09.868000	121.37.198.224	TCP	.182	74	60706	60706 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1463817031 TSecr=0 WS=128
2021-10-28 07:17:10.844000	121.37.198.224	TCP	.4	74	1327	1327 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1463817013 TSecr=0 WS=128
2021-10-28 07:17:10.848000	106.14.80.233	TCP	.5	74	4965	4965 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707837 TSecr=0 WS=128
2021-10-28 07:17:10.875000	106.14.80.233	TCP	.132	74	62811	62811 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707836 TSecr=0 WS=128
2021-10-28 07:17:10.887000	106.14.80.233	TCP	.182	74	36751	36751 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1436 SACK_PERM=1 TSval=1532707840 TSecr=0 WS=128
2021-10-28 07:17:11.049000	117.78.48.135	TCP	.31	74	45910	45910 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1462057116 TSecr=0 WS=128
2021-10-28 07:17:11.100000	139.159.199.227	TCP	.103	74	26251	26251 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790711 TSecr=0 WS=128
2021-10-28 07:17:11.100000	139.159.199.227	TCP	.103	74	26255	26255 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790711 TSecr=0 WS=128
2021-10-28 07:17:11.124000	139.159.199.227	TCP	.103	74	26267	26267 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790714 TSecr=0 WS=128
2021-10-28 07:17:11.124000	139.159.199.227	TCP	.103	74	26271	26271 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790716 TSecr=0 WS=128
2021-10-28 07:17:11.124000	139.159.199.227	TCP	.103	74	26283	26283 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790726 TSecr=0 WS=128
2021-10-28 07:17:11.125000	139.159.199.227	TCP	.103	74	26287	26287 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790728 TSecr=0 WS=128
2021-10-28 07:17:11.125000	139.159.199.227	TCP	.103	74	26295	26295 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790730 TSecr=0 WS=128
2021-10-28 07:17:11.126000	139.159.199.227	TCP	.103	74	26291	26291 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790729 TSecr=0 WS=128
2021-10-28 07:17:11.128000	139.159.199.227	TCP	.103	74	26299	26299 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790733 TSecr=0 WS=128
2021-10-28 07:17:11.161000	139.159.199.227	TCP	.197	74	50227	50227 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1387790728 TSecr=0 WS=128
2021-10-28 07:17:11.209000	121.37.198.224	TCP	.219	74	14773	14773 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1463817259 TSecr=0 WS=128

» **DNS NXDOMAIN flood attacks within the MAN, evading carriers' cloud mitigation**

In February 2022, the DNS system of a Chinese financial enterprise suffered continuous DNS query flood attacks, which used low-rate DNS NXDOMAIN flood attacks for an obvious effect. In this attack incident, the source of the attack traffic and the DNS server are on the same MAN. In this way, the attack traffic couldn't be detected by the backbone mitigation resource pool, evading carriers' cloud mitigation.

» **Low-rate large resource request attacks, consuming outbound network bandwidth and evading defense**

In November 2022, short-lived DDoS attacks were launched on the portal website of a Chinese financial enterprise. The attacks consisted of high-rate combo SYN floods and low-rate large resource requests.

» **Low-rate fixed large resource request attacks**

As shown in the following figure, low-rate fixed large resource request attacks involve two types of large resources: images and large static pages. Some garbled characters are displayed in the obtained attack packets because the Chinese characters contained in the URL are forcibly converted to ASCII characters.

Obtained Packets of a Low-Rate Fixed Large Resource Request Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2022-11-11 06:50:28.383592	36.62.187.93	HTTP	.211	231	23530	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:29.495875	111.225.149.154	HTTP	.211	797	26568	GET /`` _,%E5%B7%A5%E8%A1%8C%E9%A3%8E%E8%B2%8C/%E5%AA%92%E4%BD%93%E7%9C%88%E5%B7%A5%E8%A1%8C/%E
2022-11-11 06:51:03.438525	185.191.171.10	HTTP	.211	809	20974	GET /_ ｰ /%E7%BD%91%E4%B8%8A%E5%80%BA%E5%88%B8/%E4%BF%A1%E6%81%AF%E6%8A%AB%E9%9C%B2/%E5%80%BA%E
2022-11-11 06:51:15.846519	223.220.36.52	HTTP	.211	231	22200	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:16.008534	110.167.159.252	HTTP	.211	231	22977	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:18.694005	125.72.224.148	HTTP	.211	231	25487	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:46.527046	220.164.148.91	HTTP	.211	231	25686	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:28.071219	110.152.96.215	HTTP	.211	231	24506	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:28.114694	124.117.145.40	HTTP	.211	231	28564	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:30.487281	49.118.94.74	HTTP	.211	231	25122	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:31.790779	49.118.204.89	HTTP	.211	231	26442	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:32.101018	124.117.56.4	HTTP	.211	231	27872	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:33.042208	218.84.130.178	HTTP	.211	231	29267	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:36.533725	120.71.203.83	HTTP	.211	231	21152	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:50.817643	54.36.148.207	HTTP	.211	285	24019	GET /.`` /%E8%B5%84%E4%BA%A7%E6%89%98%E7%AE%A1/ HTTP/1.1

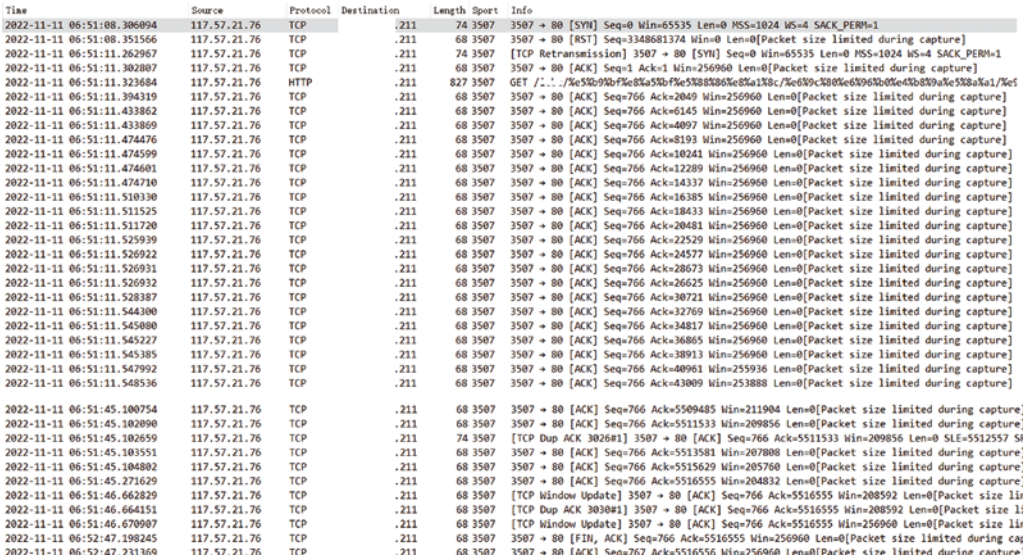
The following figure shows the obtained attack packets of typical low-rate attacks. As shown in the figure, an attack source IP address generates only three large resource requests within 1 minute, and the attack rate is far lower than 1 rps.

Obtained Packets of a Low-Rate Large Resource Request Attack (Requesting Images)

Time	Source	Protocol	Destination	Length	Sport	Info
2022-11-11 06:50:33.988664	60.173.187.140	HTTP	.211	231	44145	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.375009	60.173.20.7	HTTP	.211	231	33055	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.390209	60.173.22.254	HTTP	.211	231	43643	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.348997	60.173.29.26	HTTP	.211	231	41006	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.292095	60.173.49.60	HTTP	.211	231	36162	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.073179	60.173.51.67	HTTP	.211	231	42307	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:57.840652	60.175.115.102	HTTP	.211	216	37436	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.468699	60.175.124.108	HTTP	.211	231	41695	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:28.178774	60.175.152.55	HTTP	.211	231	51665	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:21.874977	60.20.32.203	HTTP	.211	216	60499	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:22.937563	60.20.32.203	HTTP	.211	216	60504	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:24.981575	60.20.32.203	HTTP	.211	216	60508	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:14.194594	60.221.209.103	HTTP	.211	364	60659	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:59.868841	60.232.177.102	HTTP	.211	216	40571	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:01.339269	60.232.177.102	HTTP	.211	216	40578	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:03.418649	60.232.177.102	HTTP	.211	216	40582	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:30.250785	60.5.255.38	HTTP	.211	364	38892	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:15.159341	60.5.255.39	HTTP	.211	364	58441	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:18.980570	61.147.210.45	HTTP	.211	364	43594	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:32.657210	61.154.186.2	HTTP	.211	216	45826	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:33.808925	61.154.186.2	HTTP	.211	216	45831	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:35.936370	61.154.186.2	HTTP	.211	216	45835	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:50:53.246342	61.162.101.179	HTTP	.211	364	23356	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:52:13.689930	61.163.112.33	HTTP	.211	364	23844	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:14.500511	61.163.112.38	HTTP	.211	364	50757	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:47.930192	61.166.225.181	HTTP	.211	216	6366	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:49.057917	61.166.225.181	HTTP	.211	216	6411	GET /do_not_delete/networkdetect.gif HTTP/1.1
2022-11-11 06:51:51.164983	61.166.225.181	HTTP	.211	216	6486	GET /do_not_delete/networkdetect.gif HTTP/1.1

As shown in the following figure, the attacking device sends an 827-byte request and receives a response (5,514,506 bytes) within 40 seconds. The generated outbound traffic bandwidth is about 1.1 Mbps. If an attacker invokes 10,000 zombies to launch attacks, about 10 Gbps outbound bandwidth is generated. This shows that even low-rate attacks can still achieve significant attack effects as long as there are a large number of zombies.

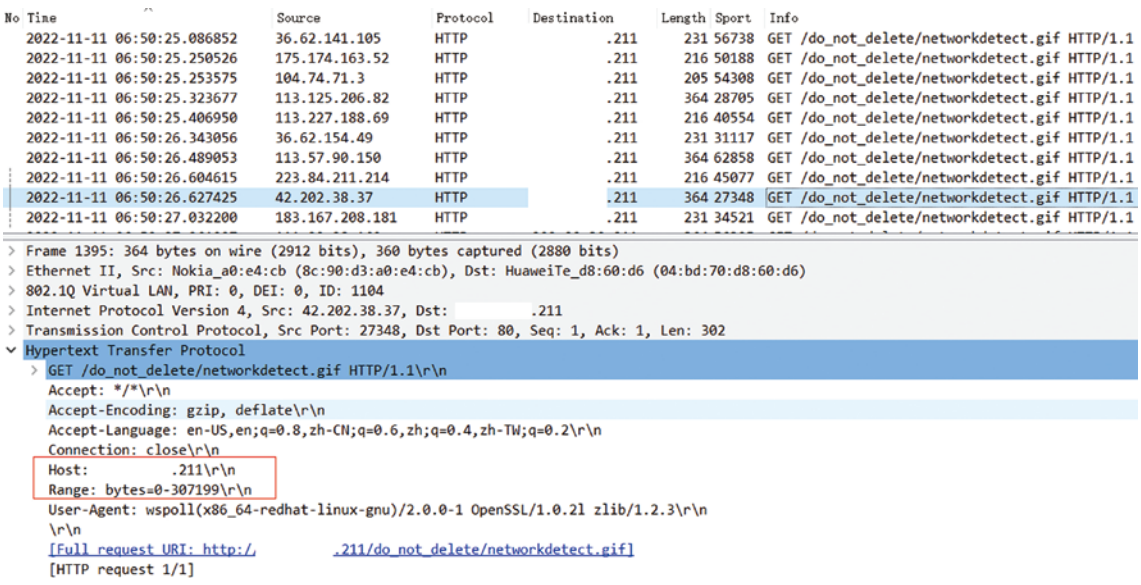
Obtained Packets of a Large Resource Complete Session Request Attack



Time	Source	Protocol	Destination	Length	Sport	Info
2022-11-11 06:51:08.306094	117.57.21.76	TCP	.211	74	3507	3507 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1024 WS=4 SACK_PERM=1
2022-11-11 06:51:08.351566	117.57.21.76	TCP	.211	68	3507	3507 → 80 [RST] Seq=3348681374 Win=0 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.262967	117.57.21.76	TCP	.211	74	3507	[TCP Retransmission] 3507 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1024 WS=4 SACK_PERM=1
2022-11-11 06:51:11.302807	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=1 Ack=1 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.323684	117.57.21.76	HTTP	.211	827	3507	GET /..%..../%e5%b9%bf%e8%a5%bf%e5%88%86%e8%a1%8c/%e6%9c%80%e6%96%b0%e4%b8%8a%e%8a%a1/%e5
2022-11-11 06:51:11.394319	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=2049 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.433862	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=6145 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.433869	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=4097 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.474476	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=8193 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.474599	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=10241 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.474601	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=12289 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.474710	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=14337 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.510330	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=16385 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.511525	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=18433 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.511720	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=20481 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.525939	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=22529 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.526922	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=24577 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.526931	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=28673 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.526932	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=26625 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.528387	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=30721 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.544300	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=32769 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.545080	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=34817 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.545227	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=36865 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.545385	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=38913 Win=256960 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.547992	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=40961 Win=255936 Len=0[Packet size limited during capture]
2022-11-11 06:51:11.548536	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=43009 Win=253888 Len=0[Packet size limited during capture]
2022-11-11 06:51:45.100754	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=5509485 Win=211904 Len=0[Packet size limited during capture]
2022-11-11 06:51:45.102090	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=5511533 Win=209856 Len=0[Packet size limited during capture]
2022-11-11 06:51:45.102659	117.57.21.76	TCP	.211	74	3507	[TCP Dup ACK 3026#1] 3507 → 80 [ACK] Seq=766 Ack=5511533 Win=209856 Len=0 SLE=5512557 SRE=5513581
2022-11-11 06:51:45.103551	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=5513581 Win=207808 Len=0[Packet size limited during capture]
2022-11-11 06:51:45.104802	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=5515629 Win=205760 Len=0[Packet size limited during capture]
2022-11-11 06:51:45.271629	117.57.21.76	TCP	.211	68	3507	3507 → 80 [ACK] Seq=766 Ack=5516555 Win=204832 Len=0[Packet size limited during capture]
2022-11-11 06:51:46.662829	117.57.21.76	TCP	.211	68	3507	[TCP Window Update] 3507 → 80 [ACK] Seq=766 Ack=5516555 Win=208592 Len=0[Packet size limited during capture
2022-11-11 06:51:46.664151	117.57.21.76	TCP	.211	68	3507	[TCP Dup ACK 3030#1] 3507 → 80 [ACK] Seq=766 Ack=5516555 Win=208592 Len=0[Packet size limited during captur
2022-11-11 06:51:46.670907	117.57.21.76	TCP	.211	68	3507	[TCP Window Update] 3507 → 80 [ACK] Seq=766 Ack=5516555 Win=256960 Len=0[Packet size limited during capture
2022-11-11 06:52:47.198245	117.57.21.76	TCP	.211	68	3507	3507 → 80 [FIN, ACK] Seq=766 Ack=5516555 Win=256960 Len=0[Packet size limited during capture]

As shown in the following figure, the attacker directly specifies the target IP address to obtain the server IP address, without performing DNS addressing. In addition, the Range option in the HTTP header is set to require the server to send the specified content, increasing server resource consumption.

Obtained Packets of a Large Resource Request Attack (Host Plaintext)



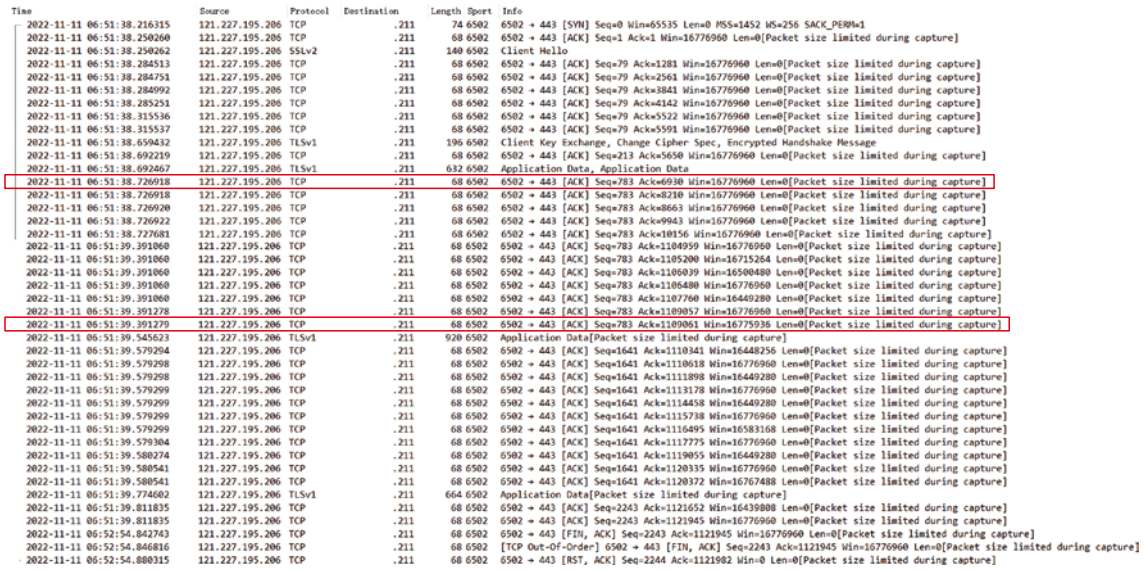
No	Time	Source	Protocol	Destination	Length	Sport	Info
	2022-11-11 06:50:25.086852	36.62.141.105	HTTP	.211	231	56738	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:25.250526	175.174.163.52	HTTP	.211	216	50188	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:25.253575	104.74.71.3	HTTP	.211	205	54308	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:25.323677	113.125.206.82	HTTP	.211	364	28705	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:25.406950	113.227.188.69	HTTP	.211	216	40554	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:26.343056	36.62.154.49	HTTP	.211	231	31117	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:26.489053	113.57.90.150	HTTP	.211	364	62858	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:26.604615	223.84.211.214	HTTP	.211	216	45077	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:26.627425	42.202.38.37	HTTP	.211	364	27348	GET /do_not_delete/networkdetect.gif HTTP/1.1
	2022-11-11 06:50:27.032200	183.167.208.181	HTTP	.211	231	34521	GET /do_not_delete/networkdetect.gif HTTP/1.1

```
> Frame 1395: 364 bytes on wire (2912 bits), 360 bytes captured (2880 bits)
> Ethernet II, Src: Nokia_a0:e4:cb (8c:90:d3:a0:e4:cb), Dst: HuaweiTe_d8:60:d6 (04:bd:70:d8:60:d6)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 1104
> Internet Protocol Version 4, Src: 42.202.38.37, Dst:        .211
> Transmission Control Protocol, Src Port: 27348, Dst Port: 80, Seq: 1, Ack: 1, Len: 302
v Hypertext Transfer Protocol
  > GET /do_not_delete/networkdetect.gif HTTP/1.1\r\n
    Accept: */*\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-US,en;q=0.8,zh-CN;q=0.6,zh;q=0.4,zh-TW;q=0.2\r\n
    Connection: close\r\n
    Host:          .211\r\n
    Range: bytes=0-307199\r\n
    User-Agent: wspoll(x86_64-redhat-linux-gnu)/2.0.0-1 OpenSSL/1.0.2l zlib/1.2.3\r\n
    \r\n
    [Full request URI: http:/,         .211/do_not_delete/networkdetect.gif]
    [HTTP request 1/1]
```

» **Low-rate encrypted large resource request attacks**

The following figure shows the obtained attack packets of typical low-rate large resource request attacks. As shown in the figure, the attack source sends three requests to the target server within 74 seconds. The first request contains 632 bytes, and the server replies with a response of 1,102,131 bytes.

Obtained Packets of a TLS Encrypted Large Resource Request Attack



Time	Source	Protocol	Destination	Length	Sport	Info
2022-11-11 06:51:38.216315	121.227.195.206	TCP	.211	74	6502	6502 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=256 SACK_PERM=1
2022-11-11 06:51:38.250260	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1 Ack=1 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.250262	121.227.195.206	SSLv2	.211	140	6502	Client Hello
2022-11-11 06:51:38.284513	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=1281 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.284751	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=2561 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.284992	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=3841 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.285251	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=4142 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.315536	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=5522 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.315537	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=79 Ack=5591 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.659432	121.227.195.206	TLSv1	.211	196	6502	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
2022-11-11 06:51:38.692219	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=213 Ack=5650 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.692467	121.227.195.206	TLSv1	.211	632	6502	Application Data, Application Data
2022-11-11 06:51:38.726918	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=6930 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.726918	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=8210 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.726920	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=8663 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.726922	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=9943 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:38.727681	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=10156 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391060	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1104959 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391060	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1105200 Win=16715264 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391060	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1106039 Win=16500480 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391060	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1106480 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391060	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1107760 Win=16449280 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391278	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1109057 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.391279	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=783 Ack=1109061 Win=16775936 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.545623	121.227.195.206	TLSv1	.211	920	6502	Application Data[Packet size limited during capture]
2022-11-11 06:51:39.579294	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1110341 Win=16448256 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579298	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1110618 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579298	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1111898 Win=16449280 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579299	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1113178 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579299	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1114458 Win=16449280 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579299	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1115738 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579299	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1116495 Win=16583168 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.579304	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1117775 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.580274	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1119055 Win=16449280 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.580541	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1120335 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.580541	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=1641 Ack=1120372 Win=16767488 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.774602	121.227.195.206	TLSv1	.211	664	6502	Application Data[Packet size limited during capture]
2022-11-11 06:51:39.811835	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=2243 Ack=1121652 Win=16439808 Len=0[Packet size limited during capture]
2022-11-11 06:51:39.811835	121.227.195.206	TCP	.211	68	6502	6502 → 443 [ACK] Seq=2243 Ack=1121945 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:52:54.842743	121.227.195.206	TCP	.211	68	6502	6502 → 443 [FIN, ACK] Seq=2243 Ack=1121945 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:52:54.846816	121.227.195.206	TCP	.211	68	6502	[TCP Out-Of-Order] 6502 → 443 [FIN, ACK] Seq=2243 Ack=1121945 Win=16776960 Len=0[Packet size limited during capture]
2022-11-11 06:52:54.880315	121.227.195.206	TCP	.211	68	6502	6502 → 443 [RST, ACK] Seq=2244 Ack=1121982 Win=0 Len=0[Packet size limited during capture]

» **Low-rate attacks through uncommon HTTP methods, evading defense**

Attackers may use HTTP methods that are not commonly used to evade defense. In November 2022, HTTP HEAD flood attacks were launched on the application layer of a financial enterprise portal website. The attacks were low-rate and had stronger attack evasion capabilities.

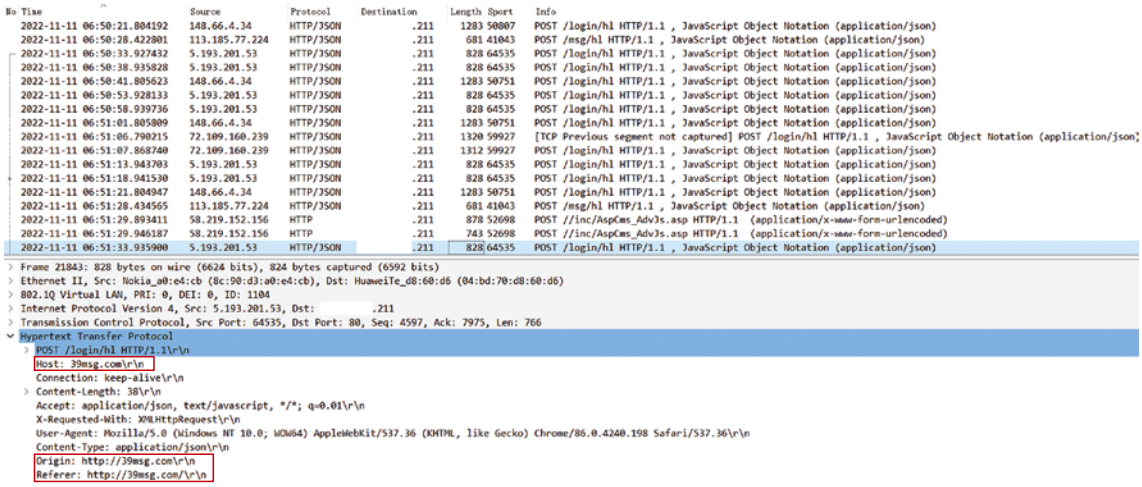
Obtained Packets of an HTTP HEAD Flood Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2022-11-11 06:50:41.086508	27.18.177.95	HTTP	.211	376	44739	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:50:42.891965	58.57.29.18	HTTP	.211	358	55920	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:50:44.612713	111.21.219.202	HTTP	.211	394	62262	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:50:55.919406	39.144.5.98	HTTP	.211	365	21135	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:50:57.426461	115.216.73.234	HTTP	.211	384	57864	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:10.040108	113.75.17.147	HTTP	.211	358	7956	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:16.185257	219.145.76.152	HTTP	.211	364	3416	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:28.036994	121.207.141.212	HTTP	.211	358	12331	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:36.748097	218.1.96.182	HTTP	.211	384	24926	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:46.389109	125.67.228.38	HTTP	.211	358	18613	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:51:55.008621	27.20.184.96	HTTP	.211	365	22319	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:03.653446	113.110.236.62	HTTP	.211	358	63568	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:12.569073	124.79.126.184	HTTP	.211	384	53395	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:13.540367	101.78.142.134	HTTP	.211	392	53637	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:16.885043	49.112.98.52	HTTP	.211	376	24429	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:23.520691	222.242.221.134	HTTP	.211	376	49334	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:26.615006	59.174.84.22	HTTP	.211	358	50530	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:33.112776	27.151.101.112	HTTP	.211	377	9099	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:36.633132	117.139.127.131	HTTP	.211	404	45412	HEAD /IEStandards.xml HTTP/1.1
2022-11-11 06:52:39.535868	60.166.92.135	HTTP	.211	384	54486	HEAD /IEStandards.xml HTTP/1.1

» **Redirected HTTP POST flood attacks, evading defense**

In November 2022, HTTP POST traffic redirected from other websites was detected in the attacks targeting the portal website of a financial enterprise. It is found that the attacker exploits the system vulnerabilities of 39msg.com to implant malicious code and redirect the traffic destined for 39msg.com to the target server. This attack technique can effectively hide the attack source.

Obtained Packets of an HTTP POST Flood Attack

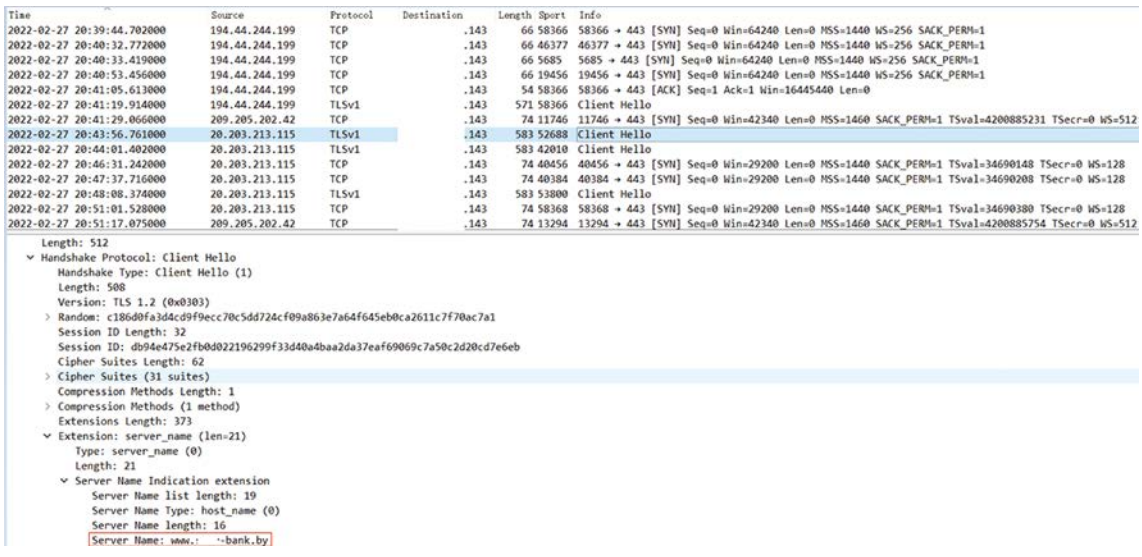


» **Low-rate incomplete TLS session attacks, consuming load balancing resources and evading defense**

In 2022, incomplete TLS session attacks on financial enterprise portals in different countries were launched. The attacks were obviously low-rate.

The following figure shows the obtained attack packets of typical low-rate attacks from a single source IP address. As shown in the figure, a maximum of three sessions can be created by an attack source IP address within 1 minute, and the session creation rate is far lower than 1 cps.

Obtained Packets of an Incomplete TLS Session Attack: Only Client Hello Packets Are Sent



» **Low-rate TLS flood attacks, consuming server resources and evading defense**

In February 2022, TLS flood attacks occurred in the large-scale cyber attacks caused by geopolitics on the financial enterprises of a country. The attacks were launched from a single source IP address and at a low rate.

The following figure shows the obtained attack packets of typical low-rate attacks from a single source IP address. As shown in the following figure, the attack source establishes multiple TCP connections with the target server and sends requests every 6 minutes.

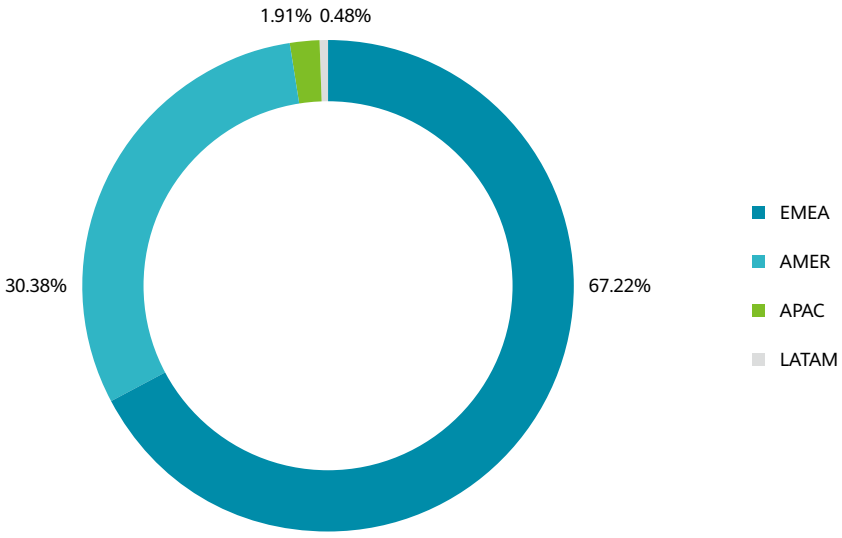
Obtained Packets of a Low-Rate Encrypted Attack

Time	Source	Protocol	Destination	Length	Sport	Info
2022-02-27 20:43:52.599000	213.110.141.238	TLSv1.2	.143	504	1035	Application Data
2022-02-27 20:49:41.466000	213.110.141.238	TLSv1.2	.143	504	1035	Application Data
2022-02-27 20:39:40.971000	213.110.141.238	TLSv1.2	.143	471	1062	Application Data
2022-02-27 20:44:53.973000	213.110.141.238	TLSv1.2	.143	471	1062	Application Data
2022-02-27 20:52:54.671000	213.110.141.238	TLSv1.2	.143	471	1062	Application Data
2022-02-27 20:43:25.848000	213.110.141.238	TLSv1.2	.143	425	1110	Application Data
2022-02-27 20:49:41.517000	213.110.141.238	TLSv1.2	.143	425	1110	Application Data
2022-02-27 20:54:05.530000	213.110.141.238	TLSv1.2	.143	425	1110	Application Data
2022-02-27 20:46:27.593000	146.70.52.21	TLSv1.2	.143	453	1174	Application Data
2022-02-27 20:40:19.752000	213.110.141.238	TLSv1.2	.143	481	1177	Application Data
2022-02-27 20:45:34.296000	213.110.141.238	TLSv1.2	.143	481	1177	Application Data
2022-02-27 20:51:08.904000	213.110.141.238	TLSv1.2	.143	481	1177	Application Data
2022-02-27 20:42:27.055000	213.110.141.238	TLSv1.2	.143	478	1189	Application Data
2022-02-27 20:49:43.685000	213.110.141.238	TLSv1.2	.143	478	1189	Application Data
2022-02-27 20:52:54.505000	98.159.226.96	TLSv1.2	.143	498	1203	Application Data
2022-02-27 20:42:20.187000	213.110.141.238	TLSv1.2	.143	463	1210	Application Data
2022-02-27 20:48:16.980000	213.110.141.238	TLSv1.2	.143	463	1210	Application Data
2022-02-27 20:53:17.134000	213.110.141.238	TLSv1.2	.143	463	1210	Application Data
2022-02-27 20:45:34.259000	37.19.197.35	TLSv1.2	.143	487	1326	Application Data
2022-02-27 20:39:44.036000	213.110.141.238	TLSv1.2	.143	459	1498	Application Data
2022-02-27 20:44:57.871000	213.110.141.238	TLSv1.2	.143	459	1498	Application Data
2022-02-27 20:50:52.497000	213.110.141.238	TLSv1.2	.143	459	1498	Application Data
2022-02-27 20:44:26.651000	45.12.26.219	TLSv1.2	.143	411	1539	Application Data
2022-02-27 20:50:31.501000	45.12.26.219	TLSv1.2	.143	411	1539	Application Data
2022-02-27 20:43:15.653000	91.231.42.248	TLSv1.2	.143	523	1542	Application Data
2022-02-27 20:50:13.125000	91.231.42.248	TLSv1.2	.143	523	1542	Application Data
2022-02-27 20:50:22.865000	91.231.42.248	TLSv1.2	.143	426	1547	Application Data
2022-02-27 20:41:28.131000	89.184.83.13	TLSv1.2	.143	520	1646	Application Data
2022-02-27 20:47:39.755000	89.184.83.13	TLSv1.2	.143	520	1646	Application Data
2022-02-27 20:41:15.280000	185.12.142.183	TLSv1.2	.143	430	1658	Application Data
2022-02-27 20:49:48.606000	185.12.142.183	TLSv1.2	.143	430	1658	Application Data

2. Attack Source Analysis

In the large-scale cyber attacks caused by geopolitics on a financial enterprise, zombies are mainly distributed in EMEA and AMER, accounting for 67.22% and 30.38% respectively.

Zombie Distribution



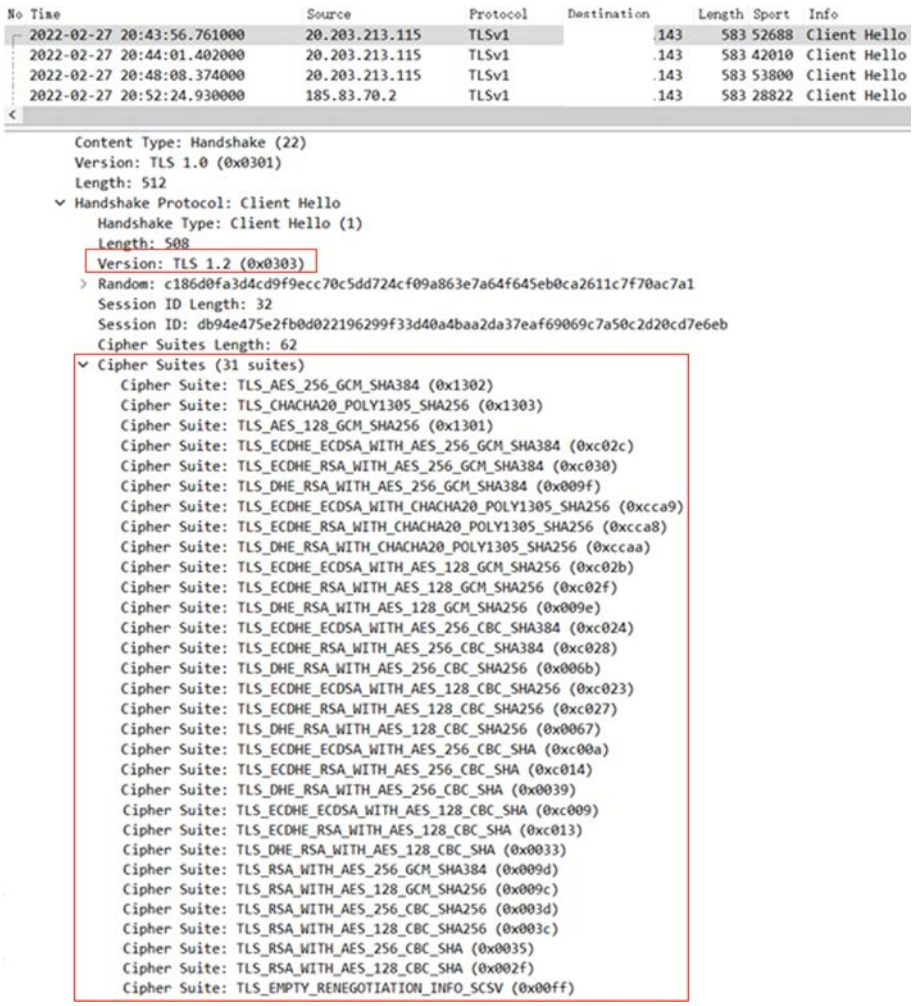
HTTPS flood attacks are mainly used. In addition, it is found that the traffic of two groups of TLS fingerprints exceeds 95% of the total.

First Group of TLS Fingerprints

No	Time	Source	Protocol	Destination	Length	Sport	Info
	2022-02-27 20:41:09.257000	13.90.89.106	TLSv1	.143	571	57078	Client Hello
	2022-02-27 20:41:27.456000	146.70.52.21	TLSv1	.143	571	61044	Client Hello
	2022-02-27 20:49:13.254000	146.70.52.21	TLSv1	.143	571	58317	Client Hello
	2022-02-27 20:53:56.145000	194.39.225.4	TLSv1	.143	583	49974	Client Hello

```
Transport Layer Security
  TLSv1 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 512
    Handshake Protocol: Client Hello
        Handshake Type: Client Hello (1)
        Length: 508
        Version: TLS 1.2 (0x0303)
      > Random: 01a5c3781da3f324481cf873c4e67fab22bb452379584373b0cf866b4b447bc6
        Session ID Length: 32
        Session ID: 1c3049ee4a117b8ef9293aafde753e1579df94c0d712d624aef15d4c1f3f710d
        Cipher Suites Length: 36
      Cipher Suites (18 suites)
          Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
          Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
          Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
          Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
          Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
          Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
          Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
          Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
          Cipher Suite: TLS_EMPTY_RENEGOTIATION_INFO_SCSV (0x00ff)
```

Second Group of TLS Fingerprints



After analysis of TLS fingerprints and attack types, it can be concluded that at least three botnets are invoked during the attack.

Attack Type / Cipher Suite	Low-Rate Incomplete TLS Session Attack	Low-Rate TLS Flood Attack
18 suites	Y	N
31 suites	Y	Y

4.2.6 DDoS Protection Suggestions for the Finance Industry

1. Summary

1) Portal websites, financial services systems, and DNS servers being the main targets, making API security a top priority

Portal websites, financial services systems, and DNS servers of large financial enterprises are the main targets of DDoS attacks. Open APIs have become main targets of DDoS attacks in terms of Internet

finance. Therefore, it is foreseeable that in the next three to five years, with the rapid development of financial digital economy, open APIs of large financial enterprises are vulnerable to cyber attacks.

2) More sophisticated, intensified, and low-rate DDoS attacks, challenging the defense system by threatening network infrastructure and applications

DDoS attacks on the finance industry mainly occur during the service settlement period. The attack targets include financial network infrastructure (bandwidth, DNS servers, and load balancing devices) and financial services systems. The attack techniques are diversified and complex, giving attackers the chance to challenge the defense system through multi-dimensional attacks.

DDoS attacks from a single source IP address are at a low rate and the attack source and targets are in the same region. This leads to high defense evasion.

Additionally, encrypted attacks targeting the finance industry have become a norm and their attack intensity has been increasing year by year. Worse yet, as encrypted attacks with tens of millions of requests per second (rps) on the Internet industry become a new normal, financial enterprises will be threatened by high-intensity encrypted attacks in the next three to five years. Using decryption methods in the defense against encrypted attacks will result in much higher defense costs than attack costs. In addition, the performance of WAFs is insufficient enough to defend against high-intensity application-layer attacks. This will ultimately make WAF performance a network development bottleneck.

2. Suggestions

1) Avoiding large resource request attacks to protect outbound link bandwidth

To reduce attacks on outbound bandwidth, CDN acceleration is recommended for images in financial services systems. Moreover, low-value large static pages on websites should not be retained for a long time and HTTP range requests should be rejected.

2) Forbidding e-banking system login through the GET method to minimize security risks

Using the HTTP GET method is insecure, which may cause CSRF attacks, sensitive information leakage, and other threats. Therefore, it is recommended that financial enterprises use the POST method for login to financial services systems.

3) Enabling domain name check to prevent access requests with non-local domain names, reducing attack risks

Both HTTP and HTTPS access requests must contain a domain name. Access requests with unauthorized domain names or raw IP addresses are denied from accessing the system, avoiding attacks launched by robots.

4) Configuring specific defense policies by service type to improve the defense success rate

Independent defense configuration is recommended for service systems that are vulnerable to attacks, such as portal websites, e-banking systems, DNS servers, and APIs. Moreover, specific defense policies can be configured based on service characteristics to improve the defense effect.

5) Selecting CDNs with security protection capabilities to avoid attack bypass

Once DDoS attacks bypass the CDN and infiltrate into the backend service system, it is more difficult to defend against them. To address this, it is recommended that financial enterprises select vendors whose CDNs can provide security protection capabilities to avoid attack bypass.

6) Upstream cloud mitigation + anti-DDoS system at the border + WAF/API gateway, improving the success rate of defending against HTTP/HTTPS application-layer attacks

Large-scale application-layer attacks challenge the performance limit of WAFs deployed by financial enterprises. Financial enterprises should quit relying on WAFs to perform single-point defense against HTTP application-layer attacks as soon as possible. Instead, they should adopt upstream cloud mitigation and deploy the anti-DDoS system at the network border to filter out high-rate application-layer attacks, protect enterprise network link bandwidth, and ensure the availability of WAFs or API gateways to filter out low-rate attacks. This helps to cope with high-intensity application-layer DDoS attacks.

7) Building a device-cloud collaborative architecture to protect enterprise link bandwidth

For financial enterprises, fast flooding attacks and carpet-bombing attacks may cause network link congestion. The limitation of flow detection technologies adopted by carriers' cloud mitigation services may lead to a 1-to-2-minute delay in defense startup. As a result, the attacks directly occupy enterprise network bandwidth and the defense fails.

The anti-DDoS system deployed at the enterprise network border provides per-packet detection, facilitating the identification of volumetric attacks threatening the enterprise network within 1 to 2 seconds. In addition, API association helps to trigger carriers' cloud mitigation services, offering effective protection to financial enterprise networks within 5 seconds.

05 Expert Opinions

Opinion 1: The intensity and frequency of DDoS attacks targeting the finance industry keep increasing, challenging the performance of WAFs. To address this, financial enterprises need to build a three-layer defense architecture consisting of carrier's upstream cloud mitigation services, an anti-DDoS system at the border of the enterprise private cloud, and WAFs to mitigate application-layer attacks.

DDoS attacks targeting the finance industry often occur during the service settlement period, meaning that these attacks are well-organized ones like the OpIcarus 20XX campaign launched by Anonymous. They attack the finance industry in an all-round manner from network infrastructure (DNS services, network bandwidth, and load balancing devices) to financial services systems with diversified and sophisticated attack techniques.

Attackers often use network-layer attacks to increase defense costs and launch application-layer attacks to target service systems. The targets of these attacks are mainly portal websites and financial services systems. If the portal websites and financial services systems are well protected, the attacks then change their targets to DNS servers.

Application-layer attacks targeting the finance industry are obviously low-rate attacks. To evade defense, attackers usually launch variable resource request attacks to challenge traditional WAFs' defense algorithms based on the source rate statistics and content matching features. In recent years, Mrps-level application-layer attacks frequently occur. In June 2022, Google Cloud tenants suffered HTTPS flood attacks[1] with 46 Mrps. Large-scale application-layer attacks directly challenge the defense performance of WAFs and the availability of enterprise network bandwidth.

To defend against large-scale application-layer attacks, upstream carriers' cloud mitigation services should be applied as the first defense line to filter out high-rate DDoS attacks and protect enterprise link bandwidth. When the bandwidth of attack traffic is suppressed within the bandwidth range of enterprises, the anti-DDoS system deployed at the border of the enterprise private cloud serves as the second defense line to filter out medium- and low-rate DDoS attacks and prevent WAFs from overloading. WAFs serve as the last defense line to intercept low-rate application-layer attacks.

Opinion 2: With APIs becoming new targets of DDoS attacks, a comprehensive DDoS protection architecture is urgently needed.

API is one of the core technologies on which mobile apps, IoT, and cloud services depend. According to statistics collected by Cloudflare, 55% of the traffic traversing the Cloudflare network is API-related[5]. API-targeted attacks have become one of the biggest threats faced by enterprises due to the large number of APIs and insufficient attention to API security. Cloudflare claims that the traffic of attacks targeting APIs intercepted by its security system has exceeded that of attacks targeting websites, indicating that APIs have become the main target of attacks[5].

According to statistics on API-targeted attack incidents collected by Huawei, APIs are also vulnerable to attacks targeting websites. In addition, due to the service characteristics of APIs, application-layer attacks targeting APIs are more complex.

To ensure the availability of open APIs, enterprises need to build a website-like layered defense architecture for APIs. That is, enterprises should apply upstream carriers' cloud mitigation services to protect the network bandwidth, deploy an anti-DDoS system at the border of the enterprise private cloud to filter out network-layer attacks within the bandwidth range of the enterprises, as well as medium- and high-rate application-layer attacks, and use WAFs or API gateways to filter out low-rate application-layer attacks.

Note:

1. **Fast flooding attack:** When volumetric attacks occur, the attack traffic increases sharply, with the peak attack traffic bandwidth reaching hundreds of Gbps within seconds.

Reference:

1. https://cloud.google.com/blog/products/identity-security/how-google-cloud-blocked-largest-layer-7-ddos-attack-at-46-million-rps
2. https://azure.microsoft.com/en-us/blog/azure-ddos-protection-2021-q3-and-q4-ddos-attack-trends/
3. https://ieeexplore.ieee.org/document/8886426
4. https://www.akamai.com/newsroom/press-release/state-of-the-internet-security-retail-attacks-and-api-traffic
5. https://www.cloudflare.com/products/api-gateway/
6. https://salt.security/api-security-trends

06 Data Source

6.1 Data Source

The data involved in this report originates from China Telecom Cybersecurity Technology Co.,Ltd., China Unicom Digital Technology Company Limited, Baidu Security, Nexusguard, Huawei Cloud, and DDoS attack–related data from Huawei's customers after authorization.

HUAWEI TECHNOLOGIES CO., LTD.
Huawei Industrial Base
Bantian Longgang
Shenzhen 518129, P. R. China
Tel: +86-755-28780808
www.huawei.com