红队实战对抗手法 提炼汇总

说明:

以下仅针对红队场景,进行了一次全面完整的实战攻击利用技术提炼汇总,针对不同渗透阶段,所可能会用到的一些具体技术都做了详尽梳理说明[包括后面也可能还会整理出一批对应的完整工具链,虽然工具不是最主要的], 由于红队不同于一般的渗透测试,强调更多的是如何搞进去, 拿到相应的目标机器权限或者实现某些特定目的,而不局限于你一定要用什么技术,或者必须通过什么途径去搞,相比传统渗透测试,红队则更趋于真实入侵活动,这种场景其实对防御者的实战对抗经验和技术深度都是比较大的挑战,所以以下的所有技术点也几乎都是完全站在这种场景和角度下去考量梳理的,另外需要说明的是,所有攻击手法都绝不是完全孤立使用的,往往很多手法在实战利用过程中都是相互灵活组合起来进行循环利用,由于绝大部分内容都是基于本人平时的一些学习和实战经验积累,加之每个人的实际渗透思路都不同,所以肯定会有遗漏的地方,也欢迎弟兄们一起来积极指正补充,个人觉得,最好的防御不是怎么去防工具,因为工具这些东西本身就是死的,稍微改下,变化下现有的规则可能马上就防不住了,且一直会处于疲于应付的被动防御状态,尤其是针对红队这种特殊场景的,你的实际对手可能都是有一定技术实力的人,所以针对每种核心的攻击技术技术展开做深入分析,直接从源头上进行防御才是最靠谱的,虽然短期这种成本代价相对较高,但长期来看,是一劳永逸的,这种沉淀下来的东西最终也会慢慢形成自己产品的核心竞争力和特色,说白点,这种对抗,本身拼的就是双方的技术实力

文章要点预览:

```
入口权限获取 [ 前期外部搜集侦察 ]
入口权限获取 [ 针对各主流 "中间件 + 开源程序 + Web 服务组件" 自身的各种已知 Nday 利用 ]
入口权限获取 [ 针对各类基础服务端口 getshell 利用 ]
入口权限获取 [ 传统钓鱼攻击利用 ]
主机安全 [ Windows & linux 系统提权利用  ]
内网安全 [ 敏感信息搜集 ]
内网安全  [ 各类敏感凭证搜集 与 窃取  ]
内网安全 [ 常用 "隧道"" / "转发"" / "代理"" 穿透手法 ]
域内网安全 [ 域渗透 ]
内网安全 [ 跨平台横向渗透 (远程执行) ]
内网安全 [ 权限维持 ]
各类常用 C2 / 渗透 框架分析
各类常用 Webshell 管理工具分析
免杀 及 各类防火墙对抗
...
```

0x01 入口权限获取 [前期侦察, 搜集阶段本身就不存在太多可防御的点,不是防御的重心]

```
绕 CDN 找出目标所有真实 ip 段
找目标各种 Web 管理后台登录口
批量抓取目标所有真实 C 段 Web banner
批量对目标所有真实 C 段 进行基础服务端口扫描探测识别
尝试目标 DNS 是否允许区域传送,如果不允许则继续尝试子域爆破
批量抓取目标所有子域 Web banner
批量对目标所有子域集中进行基础服务端口探测识别
批量识别目标 所有存活 Web 站点的 Web 程序指纹 及其详细版本
从 Git 中查找目标泄露的各类 敏感文件 及 账号密码,偶尔甚至还能碰到目标不小心泄露的各种云的 "AccessKey"
从网盘 / 百度文库 中查找目标泄露的各类 敏感文件 及 账号密码
从各第三方历史漏洞库中查找目标曾经泄露的 各种敏感账号密码 [ 国内目标很好使 ]
目标 Svn 里泄露的各类 敏感文件
网站目录扫描 [ 查找目标网站泄露的各类敏感文件, 网站备份文件, 敏感配置文件, 源码 , 别人的 webshell, 等等等...]
目标站点自身在前端代码中泄露的各种敏感信息
fofa / shodan / bing / google  hacking 深度利用
搜集目标 学生学号 / 员工工号 / 目标邮箱 [ 并顺手到各个社工库中去批量查询这些邮箱是否曾经泄露过密码 ]
目标自己对外提供的各种 技术文档 / wiki 里泄露的各种账号密码及其它敏感信息
目标微信小程序
分析目标 app Web 请求
借助 js 探针搜集目标内网信息
想办法混入目标的各种 内部 QQ 群 / 微信群
分析目标直接供应商 [尤其是技术外包]
根据前面已搜集到的各类信息制作有针对性的弱口令字典
目标所用 Waf 种类识别 与 绕过
   BypassWAF 文件上传 / 读取 / 下载
   BypassWAF Sql 注入
   BypassWAF RCE
   BypassWAF 各类 Java Web 中间件已知 Nday 漏洞利用
   BypassWAF Webshell 免杀

其它 , 待续...
```

0x02 入口权限获取 [此阶段,主要针对各主流 "中间件 + 开源程序 + Web 服务组件" 自身的各种已知 Nday 漏洞利用, 也是整个外部防御的重心("重中之重")]

如下已按 "实际攻击利用的难易程度" 及 "获取到的 shell 权限高低" 为标准进行了详细排序
此处完全以实战利用为导向,故,仅仅只挑选了一些相对会经常遇到的,且实战中确实能有效协助快速 getshell 的 "中间件" , "开源程序" 及 "web 组件"

1) 针对各类 Java 中间件的 各种已知 Nday 漏洞利用

不同于其它脚本类 web 程序,Java 的运行权限通常都比较高,甚至大部分都是直接用 root / administrator / system 权限在跑,所以拿到的 shell 权限一般也非常高,通常都直接是服务器权限,所以,尤其是在各种日常红队这种场景中,入侵者一般也都会首选这些点,并以此为突破口来获取一个稳定的跳板机入口权限,关于到底哪些行业特别爱用哪些中间件,这些也应该都是有事先分析梳理汇总好的

```
Struts2
   Struts2-005
   Struts2-008
   Struts2-009
   Struts2-013
   Struts2-016 [ 实际上,很多都老系统都漏补了这个洞,成功率较高 ]
   Struts2-019
   Struts2-020
   Struts2-devmode
   Struts2-032
   Struts2-033
   Struts2-037
   Struts2-045
   Struts2-046
   Struts2-048
   Struts2-052
   Struts2-053
   Struts2-057

Weblogic
   CVE-2019-2725
   CVE-2019-2729
   CVE-2018-3191
   CVE-2018-2628
   CVE-2018-2893
   CVE-2018-2894
   CVE-2017-3506
   CVE-2017-10271
   CVE-2017-3248
   CVE-2016-0638
   CVE-2016-3510
   CVE-2015-4852
   CVE-2014-4210
   SSRF
   控制台弱口令,部署 webshell

Jboss
   CVE-2015-7501
   CVE-2017-7504
   CVE-2017-12149
   未授权访问,部署 webshell
   控制台弱口令,部署 webshell

Wildfly [ jboss 7.x 改名为 wildfly ]
   控制台弱口令,部署 webshell

Tomcat
   CVE-2016-8735
   CVE-2017-12615 [ readonly 实际设为 true 的情况较少]
   控制台弱口令,部署 webshelll [ 注: 7.x 版本后,默认加了防爆机制 ]

Jekins
   CVE-2018-1999002 [ 任意文件读取 ]
   未授权访问,任意命令执行
   控制台弱口令,任意命令执行
```

```
ElasticSearch
   CVE-2014-3120 [ 专门针对老版本(无沙盒)RCE ]
   CVE-2015-1427 [ Groovy RCE ]
   CVE-2015-3337 [ 任意文件读取 ]
   未授权访问,敏感信息泄露

RabbitMQ
   弱口令

Glassfish
   任意文件读取 [ 低版本 ]
   控制台弱口令,部署 webshell

IBM Websphere
   Java 反序列化
   控制台弱口令,部署 webshell

Axis2
   任意文件读取
   目录遍历

Apache ActiveMQ
   未授权访问,PUT 任意写[ 针对老版本 ]
   CVE-2015-5254

Apache Solr
   CVE-2017-12629
   CVE-2019-0193 [ Apache Solr 5.x - 8.2.0 ]

Apache Zookeeper
   未授权访问,敏感信息泄露

Apache Shiro 反序列化
fastjson <= 1.2.47 反序列化利用

针对各类 Windows php 集成环境 [ 由于此类环境拿到的 Webshell 权限相对较高,所以,通常也是红队人员的首选突破口 ]
   AppServ
   Xampp
   宝塔
   PhpStudy
```

2）针对各类开源程序的 已知 Nday 漏洞利用

```
Dedecms 后台弱口令,系列已知 nday 漏洞利用
thinkphp 5.x 后台弱口令,系列已知 nday 漏洞利用
phpcms 后台弱口令,系列已知 nday 漏洞利用
ecshop 后台弱口令,系列已知 nday 漏洞利用
Metinfo 后台弱口令,系列已知 nday 漏洞利用
Discus 后台弱口令,系列已知 nday 漏洞利用
帝国 cms 后台弱口令,系列已知 nday 漏洞利用
PhpMyAdmin 后台弱口令,系列已知 nday 漏洞利用
Wordpress 后台弱口令,系列已知 nday 漏洞利用
Joomla 后台弱口令,系列已知 nday 漏洞利用
Drupal CVE-2018-7600, 后台弱口令,系列已知 nday 漏洞利用
...
```

3）针对其它各类 Web 组件的 已知 Nday 漏洞利用

```
IIS 6.0 RCE
   短文件漏洞
   PUT 任意写
   Webdav RCE CVE-2017-7269

禅道项目管理系统
   SQL 注入
   文件读取
   远程执行

通达 OA
   SQL 注入
   任意上传

Zimbra [ XXE + SSRF => RCE ]
   CVE-2013-7091
   CVE-2016-9924
   CVE-2019-9670

Citrix
   CVE-2019-19781

Jumpserver
   身份验证绕过

Zabbix
   CVE-2017-2824
   SQL 注入 [ 2.0 老版本 ]
   控制台弱口令,敏感机器信息泄露

Cacti
   低版本 SQL 注入
   控制台弱口令

Nagios
   CVE-2016-9565
   控制台弱口令

Webmin RCE
   CVE-2019-15107

PHPMailer
   CVE-2016-10033

泛微 OA 远程代码执行
金蝶 OA SQL 注入
Coremail 敏感文件泄露
UEditor 任意文件上传
OpenSSL 心脏滴血[Heartbleed] 抓 明文账号密码
破壳漏洞 [Shellshock] RCE
```

4）各种常规 Web 漏洞利用 [注：有些漏洞在不审代码的情况下其实是很难有效盲测到的]

```
后台弱口令
SSRF
sql 注入
越权
命令 / 代码执行 / 反序列化
任意文件上传 / 下载 / 读取
包含
XSS [ 实际上,此类漏洞只有在针对某些特定邮箱,手里有浏览器 0day 时价值才会比较大,红队这种场景其实并不是非常非常致命,因为也没有太多时间精力搞这些 ]
业务逻辑漏洞
```

5）针对各类边界网络设备的各种利用,主要是 Web 管理控制台登录弱口令 及 各类已知 nday 攻击利用

```
Pulse Secure VPN
   CVE-2019-1151 [ 任意文件读取 ]

Fortinet VPN
   CVE-2018-13379 [ 文件读取 ]

Sangfor Vpn RCE
```

0x03 入口权限获取 [专门针对各类基础服务端口的各种 getshell 利用 , 是防御重点 ("重中之重")]

此处仅仅只挑选了一些实战中真正能协助快速 getshell 的服务,其它的一些相对边缘性的服务均未提及
同样,已按 "实际攻击利用的难易程度" 及 "获取到的 shell 权限高低" 为标准进行了详细排序
如下,就每个端口的具体攻击利用方式,进行了简要说明

```
Mssql         [ 默认工作在 tcp 1433 端口, 弱口令, 敏感账号密码泄露, 提权, 远程执行, 后门植入 ]
SMB           [ 默认工作在 tcp 445 端口, 弱口令, 远程执行, 后门植入 ]
WMI           [ 默认工作在 tcp 135 端口, 弱口令, 远程执行, 后门植入 ]
WinRM         [ 默认工作在 tcp 5985 端口, 此项主要针对某些高版本 Windows, 弱口令, 远程执行, 后门植入 ]
RDP           [ 默认工作在 tcp 3389 端口, 弱口令, 远程执行, 别人留的 shift 类后门 ]
SSH           [ 默认工作在 tcp 22 端口, 弱口令, 远程执行, 后门植入 ]
ORACLE        [ 默认工作在 tcp 1521 端口, 弱口令, 敏感账号密码泄露, 提权, 远程执行, 后门植入 ]
Mysql         [ 默认工作在 tcp 3306 端口, 弱口令, 敏感账号密码泄露, 提权(只适用于部分老系统) ]
REDIS         [ 默认工作在 tcp 6379 端口, 弱口令, 未授权访问, 写文件(webshell,启动项,计划任务), 提权(比如,服务以高权限运行便可达到此效果) ]
POSTGRESQL    [ 默认工作在 tcp 5432 端口, 弱口令, 敏感信息泄露 ]
LDAP          [ 默认工作在 tcp 389 端口, 未授权访问, 弱口令, 敏感账号密码泄露 ]
SMTP          [ 默认工作在 tcp 25 端口, 服务错误配置导致的用户名枚举漏洞, 弱口令, 敏感信息泄露 ]
POP3          [ 默认工作在 tcp 110 端口, 弱口令, 敏感信息泄露 ]
IMAP          [ 默认工作在 tcp 143 端口, 弱口令, 敏感信息泄露 ]
Exchange      [ 默认工作在 tcp 443 端口, 专门针对各个接口的弱口令爆破 eg: Owa,ews,oab,AutoDiscover,Microsoft-Server-ActiveSync, pth 脱邮件, 敏感信息泄露 ... ]
VNC           [ 默认工作在 tcp 5900 端口, 弱口令 ]
FTP           [ 默认工作在 tcp 21 端口, 弱口令, 匿名访问/可写, 敏感信息泄露 ]
Rsync         [ 默认工作在 tcp 873 端口, 未授权, 弱口令, 敏感信息泄露 ]
Mongodb       [ 默认工作在 tcp 27017 端口, 未授权, 弱口令 ]
TELNET        [ 默认工作在 tcp 23 端口, 弱口令, 后门植入 ]
SVN           [ 默认工作在 tcp 3690 端口, 弱口令, 敏感信息泄露 ]
JAVA RMI      [ 默认工作在 tcp 1099 端口, 可能存在反序列化利用 ]
CouchDB       [ 默认工作在 tcp 5984 端口, 未授权访问 ]
```

0x04 入口权限获取 [传统钓鱼攻击利用 , 实际红队场景中用的非常频繁 , 细节非常多，此处不一一列举 , 是防御重点]

枚举有效的目标邮箱用户名 , 目标邮箱弱口令 , 伪造发信人
第一种,直接给目标发送各种常规木马信 , 传统宏利用 , 恶意捆绑，exe[zip]，lnk，chm，自解压，木马链接，远程模板注入，OLE，CVE-2017-11882 [利用漏洞触发]...
第二种,给目标发送各种钓鱼链接,比如，利用各种目标登录口的钓鱼页面来窃取各种内网账号密码 , Vpn , Mail , OA , Net ntlm hash...境外 ISP 公网过滤 SMB 流量

针对不同行业一般也都会事先准备好各种各样的针对性的发信话术模板,以此来提到实际发信成功率

0x05 主机安全 [提权利用 , 是防御重点]

以下只单独挑了一些在 通用性，稳定性，易用性，实际成功率 都相对较好的洞 和 方式 其它的一些"边缘性"的利用都暂未提及

```
Windows 系统漏洞 本地提权 [ 成功的前提是,保证事先已做好各种针对性免杀 ]
BypassUAC [ win7 / 8  / 8.1 / 10 ]  [重点]
MS14-058[KB3000061]                 [重点]
MS14-068[KB3011780]                 [重点]
ms15-051[KB3045171]                 [重点]
MS15-077[KB3077657]                 [重点]
MS16-032[KB3124280]                 [重点]
ms16-075                            [重点]
MS16-135[KB3199135]                 [重点]
MS17-010[KB4013389]                 [重点]
cve-2019-0708                       [重点]
CVE-2019-0803                       [重点]
CVE-2019-1322 & CVE-2019-1405       [重点]


linux 内核漏洞 本地提权 [ linux-exploit-suggester  ]
CVE-2016-5195                       [重点]
CVE-2017-16995
CVE-2019-13272


利用各类第三方服务 / 软件工具提权
Mssql                               [重点]
Oracle                              [重点]
Mysql
各类第三方软件 dll 劫持               [重点]
suid 权限
计划任务
各种错误服务配置
```

0x06 内网安全 [敏感信息搜集 , 是防御重点, 可在此项严格限制各种系统内置命令执行]

搜集当前已控"跳板机"的各类敏感信息 [注: 如下某些操作肯定是需要事先自己想办法先拿到管理权限后才能正常进行的,此处不再赘述]

查看当前 shell 权限 及 详细系统内核版本
获取当前系统的 详细 ip 配置,包括 所在域, ip, 掩码, 网关, 主备 dns ip
获取当前系统最近的用户登录记录
获取当前用户的所有命令历史记录 [主要针对 linux,里面可能包含的有各类敏感账号密码,ip,敏感服务配置...]
获取本机所有 服务 / 进程 [包括各个进程的详细权限,也包括目标系统中的可疑恶意进程(有可能是同行的马)] / 端口 / 网络连接 信息
获取本机所用杀软 / 监控种类 [后续好针对性的做免杀]
获取本机 rdp / ssh 端口开启状态 及 其默认端口号
获取本机所有用户的 rdp 外连记录
获取本机的所有 SSH 登录记录
获取当前系统所有登录成功的日志 [针对 windows]
获取本机所有已安装软件的详细列表 [主要为抓密码,提权,留后门做准备]
获取本机各个浏览器中保存的 所有书签页 及 历史浏览记录
获取当前用户创建的所有计划任务列表 及 计划任务所对应的执行脚本内容 [有些执行脚本中很可能存的有各种连接账号密码]
获取当前用户 桌面 及 回收站 里的所有文件列表
获取当前系统的所有存在 suid 权限的二进制程序
获取当前系统代理 [ip & 端口]
获取当前系统所有的自启动注册表项值
获取当前系统的所有 ipc 连接 及 已启用共享
获取当前系统的所有挂载[mount]
获取当前系统的防火墙状态
获取当前系统所有分区/盘符及其详细使用情况
获取本机的累计开机时长
获取本机 arp / dns 缓存
获取当前机器环境变量 [主要想看看目标机器上有无 python,jdk,ruby...等语言的执行环境,后期可设法利用]
获取当前系统所有本地用户及组列表
获取当前系统 host 文件内容
获取当前机器硬件设备信息[主要为判断当前机器是否为虚拟机]
远程截屏捕捉目标用户敏感操作

[针对以上单机信息搜集的防御,由于上述大部分的搜集动作都是基于系统内置工具和接口,故,可完全依靠 EDR 来实时捕捉各类敏感进程上报恶意操作]

利用当前已控 "跳板机", 分析目标内网大致拓扑 及 所有关键性业务机器分布

批量抓取内网所有 windows 机器名 和 所在 "域" / "工作组名" [smb 探测扫描]
针对内网的各种高危敏感服务扫描定位 ["安全" 端口扫描 (在尽量不触发对方防护报警拦截的情况下进行各种常规服务探测识别)]
内网批量 Web Banner 抓取,获取关键目标业务系统如下
  内网各种文件[共享]服务器
  内网各类 web 服务器 [可用于后期留入口]
  内网各类数据库服务器
  内网邮件服务器 [可用于后期留入口]
  内网 Vpn 服务器 [可用于后期留入口]
  内网各类常规资产状态监控服务器,eg: zabbix,nagios,cacti...
  内网各类防护的主控端,比如,防火墙,EDR,态势感知 产品的 web 主控端...
  内网日志服务器
  内网补丁服务器
  内网各类 OA,ERP,CRM,SRM,HR 系统...
  内网打印服务器
  内网 MES 系统
  内网虚拟化服务器 / 超融合平台 [Vmware ESX]
  内网堡垒机...
  内网运维,研发 部门员工的机器
  内网路由,交换设备...
  等等等...

针对以上的各种内网探测扫描,其实在流量上都会有非常清晰的表现,通过在一些关键节点设备/服务器上部署探针搜集流量
再配合大数据关联分析查找各种敏感特征,理论上是相对容易发现各类扫描探测痕迹的

针对各类已知系统高危 RCE 漏洞的批量探测识别与利用

MS08-067 [其实,某些特殊行业的系统可能非常老,极少更新,故,还是有存在的可能]
MS17-010
CVE-2019-0708

其实针对此类漏洞的攻击利用识别,就比较直白了,通过深入分析每种漏洞在实际攻击利用过程所产生的一些典型 流量特征 和 系统日志即可大致判断

0x07 内网安全 [各类敏感凭证搜集 与 窃取]

主动密码搜集 [注:如下某些操作肯定是需要事先自己想办法先拿到管理权限或者在指定用户权限下才能正常进行的,此处不再赘述, 此项, 非防御重点, 因为压根也不好防]

```
批量抓取当前机器上的 "各类基础服务配置文件中保存的各种账号密码",比如,各种数据库连接配置文件,各类服务自身的配置文件(redis,http basic...)...
抓取当前系统 "注册表中保存的各类账号密码 hash" [ Windows ]
抓取当前系统所有 "本地用户的明文密码/hash" [ Windows & linux ]
抓取当前系统的所有 "用户 token" [ Windows ]
抓取 "windows 凭据管理器中保存的各类连接账号密码"
抓取 "MSTSC 客户端中保存的所有 rdp 连接账号密码"
抓取各类 "VNC 客户端工具中保存的连接密码"
抓取 "GPP 目录下保存的各类账号密码" [ 包括组策略目录中 XML 里保存的密码 hash 和 NETLOGON 目录下的某些脚本中保存的账号密码 ]
抓取各类 "SSH 客户端工具中保存的各种 linux 系统连接账号密码", SecureCRT,Xshell,WinSCP,putty
抓取各类 "浏览器中保存的各种 web 登录密码" , Chrome [360 浏览器], Firefox , IE , QQ 浏览器
抓取各类 "数据库客户端工具中保存各种数据库连接账号密码", Navicat, SSMS[ MSSQL 自带的客户端管理工具,里面也可能保存的有密码(没记错的话,应该是加密后的 base64) ]
抓取各类 "数据库表中保存的各类账号密码 hash"
抓取各类 "FTP 客户端工具中保存的各种 ftp 登录账号密码", filezila, xftp...
抓取各类 "邮件客户端工具中保存的各种邮箱账号密码", forxmail, thunderbird...
抓取各类 "SVN 客户端工具中保存的所有连接账号密码及项目地址"
抓取各类 "VPN 客户端工具中保存的各种 vpn 链接账号密码"
想办法 "控制目标 运维管理 / 技术人员 的单机,从这些机器上去搜集可能保存着各类敏感网络资产的账号密码表",eg: *.ls,*.doc,*.docx, *.txt....
...
```

被动密码搜集 [注: 某些操作肯定是需要事先自己想办法先拿到管理权限后才能正常进行的, 此处不再赘述 , 是防御重点]

```
Windows SSP [ 持久化 / 内存 ]
Hook PasswordChangeNotify [ 持久化 / 内存 ]
OWA 登录账号密码截获
截获 mstsc.exe 中输入的 rdp 连接账号密码
linux 别名记录利用
本机密码嗅探 [http,ftp,pop3...]
传统键盘记录
windows 蓝屏技巧 [ 此操作主要为应对不时之需 ]
...
```

Hash 爆破

```
Hashcat
```

0x08 内网安全 [内网常用 "隧道"" / "转发"" / "代理"" 穿透手法 提炼汇总 , 是防御重点]

```
出网流量刺探          [ 比如,http,dns,以及一些穿透性相对较好的 tcp 端口... 这种操作一般都会配合 wmi,smb,ssh 远程执行,在内网批量快速识别出能出网的机器 ]
常规 http 脚本代理    [ abptts,Neo-reGeorg,reGeorg,tunna,reduh...这些公开脚本实战中多多少少都会有些问题,还需要根据自己的实际目标环境深度改进才行 ]
SSH 隧道              [ 加密端口转发,socks 实战用途非常灵活,此处不细说 ]
Rdp 隧道
反向 socks5           [ nps, frp, ssf, CobaltStrike(socks4a & rportfwd ), sscoks ... 工具基本都不免杀了,需要自行处理 ]
正反向 tcp 端口转发   [ 非常多,就不一一列举, eg: nginx,netsh,socat,ew....  ]
dns 加密隧道
Web 端口复用

[ 需要明白的是,红队场景中,入侵者为了尽可能躲避各种检测设备的流量解析,很多此类工具都会采用各种各样的方式来加密传输流量,以此来保证自己有更强的穿透性 ]
```

0x09 域内网安全 [域内常用攻击手法 提炼汇总 (域渗透), 是防御重点]

针对当前域的一些常规信息搜集 [其实,现实中, 只需要一个 BloodHound & Pingcastle 足矣, 就是工具需要自行事先免杀好]

```
获取当前域内的完整域管列表
获取当前域内的所有域控机器名列表
获取当前域内的所有 DNS 服务器机器名列表
获取当前域内的所有 SPN
获取当前域内的所有 OU
获取当前域内的所有用户 & 用户组列表
获取当前域信任关系
获取当前域内所有机器的开机时间
获取当前域内网段及 web 站点
获取当前域内策略[主要为了解密码登录策略]
获取当前域林
.......
```

快速获取目标域控权限的一些常规手法

```
搜集 GPP 目录 [ 其中可能保存的有域账号密码,不仅仅是存在 XML 里的那些,NETLOGON 目录中的某些脚本同样也可能保存有账号密码 ]
服务票据 hash 破解,"尤其是域管用户的" [kerberoast]
批量对域用户进行单密码尝试 [ 喷射,利用 ADSI 接口,日志 id 4771 ]
Kerberos 委派利用
爆破 LDAP
Exchange 特定 ACL 滥用
SSP 截获关键服务器登录密码
利用各类基础服务在内网快速 getshell [ 弱口令, 各类 JAVA 中间件已知 Nday 漏洞, 常规 Web 漏洞... ],在内网循环抓各类密码,直至
   抓到域管密码
   抓到域管令牌
DNSAdmin 组成员滥用 [ 加载执行恶意 dll ]
LAPS
MS14-068 [ 如今实际中已很少遇到了 ]
LLMNR/NBNS 欺骗  + SMB relay [ 真实实战中其实用的并不多 ]
```

域内后渗透敏感信息搜集分析

```
获取所有 DNS 记录
导出当前域的完整 LDAP 数据库
提取当前域的 ntds.dit [ 域内账号密码数据库 ]
   Dcsync 同步
   Volume Shadow Copy Service
```

域内指定用户登录 ip 定位

```
利用 OWA 登录日志
利用域控服务器登录日志
指定服务银票 [Silver Ticket]
除此之外,就是下面的各类常规横向手法
```

域内指定用户机器定向控制技巧

```
绑定用户登录脚本
利用 GPO 下发 [ 实际上 GPO 能做的事情还非常非常多 ]
PTT [ 票据传递 ]
```

针对域管的各种权限维持技巧

```
金票
Skeleton Key
DSRM 密码同步
OWA 后门
...
```

域内 Exchange 邮件数据脱取

```
利用 Ews 接口通过 PTH 的方式脱邮件
```

0x10 内网安全 [跨平台横向渗透（远程执行），是防御重点（ "重中之重" ）]

从 Windows 平台 横向至 Windows 平台 [注：以下某些远程执行方式，即可直接用明文账号密码 亦可 基于 pth 来进行，不局限]

```
远程服务管理 [ SCM ]
远程创建执行计划任务 [ Scheduled Tasks ]
WMI 远程执行 [ WMI ]
针对高版本 Windows 的 WinRM 远程执行
DCOM 远程执行 [ 需要目标 Windows 机器事先已关闭防火墙 ]
高版本 RDP 远程执行
利用 MSSQL 数据库存储过程来变相远程执行
利用 Oracle 数据库存储过程来变相远程执行
SMB [ PTH (hash 传递) ]
RDP[MSTSC] 反向渗透 [ 即可用于突破某些隔离，亦可通过云(Windows vps)直接反控到目标管理员个人机 ]
利用补丁服务器下发执行
利用 EDR 主控端定向下发执行
...
```

从 Windows 平台 横向至 *inux 平台

```
plink 或者 基于 Windows SSH 库自行开发各种远程执行小工具
```

从 *inux 平台 横向至 Windows 平台

```
一般都会将 impacket 套件中的各个常用 py 脚本事先直接打包成可执行文件，然后丢到目标 linux 系统中去执行,如下
wmiexec_linux_x86_64
smbexec_linux_x86_64
psexec_linux_x86_64
atexec_linux_x86_64
dcomexec_linux_x86_64

另外,还有一些基于 go 的工具,同样也可以编译成可执行文件之后再丢上去执行
```

从 *inux 平台 横向至 *inux 平台

```
linux 自带的 ssh 客户端工具套件，默认就可以用来进行远程执行
```

各类远程下载

```
Wget [ win & linux]
curl [ win & linux ]
```

之所以没着重提以下这些系统内置远程下载执行工具,主要还是因为事先已经明确知道,某些杀软下它肯定会被拦截,所以事先就直接把它弃用了,尤其针对红队这种场景,这些东西根本不在乎多,有一个能用好用的即可

```
CertUtil.exe
Bitsadmin.exe
Regsvr32.exe
Rundll32.exe
Powershell.exe
......
```

0x11 内网安全 [权限维持 , 是防御重点] [注: 有些细节此处并未展开详细说明]

边界入口权限维持 [临时]

```
OWA 登录口 [ 账号密码,webshell ]
VPN 登录口 [ 账号密码,shell ]
其他 MAIL 登录口 [ 账号密码 ]
边界 Web 服务器 [ Webshell 驻留 ]
边界路由交换设备 [ 账号密码,shell ]
...
```

Windows 单机系统维持 [临时]

```
系统计划任务 [ 高权限 / 低权限 ]
常规注册表自启动项 [ 用户权限 / system 权限 ]
Mssql 存储过程 [ 继承服务权限 ]
WMI
Winlogon
CLR
Logon Scripts
MruPidlList
Mof
传统远控
...
```

linux 单机系统维持 [临时]

```
Patch SSH
替换 so [ PAM,Nginx,Rsync  ... ]
系统计划任务
传统应用层远控
驱动层远控
...
```

0x12 各类常用 C2 / 渗透 框架分析

```
CobaltStrike
   beacon 逆向

Metasploit
Empire
......
```

0x13 各类常用 Webshell 管理工具分析

```
菜刀     caidao20160622
冰蟹     Behinder_v2.0.1
蚁剑     AntSword
......
```

0x14 免杀 及 各类防火墙对抗

静态

常规混淆

手动混淆,有源码的情况下,尝试逐个替换可能是关键特征字符串的 命名空间名, 函数名, 变量名, 字符串 等等等....

工具混淆,针对各种语言的专业混淆工具 [有商业版], 最好的方式还是尝试自己写混淆工具

加壳

一些常用公开壳的实际效果可能并不是太好 [也有商业壳],最好的方式还是尝试自己写壳,就是成本较高

动态

反射

shellcode 加密执行

.....

流量

域前置[利用大厂 cdn]

第三方公共邮箱上线

第三方网盘上线

第三方社交网站上线

第三方匿名社交工具上线 [eg: tg 机器人,tor...]

...

By klion
2020.2.16