**Kerberos协议的滥用**

**微软的活动目录默认使用Kerberos处理认证请求。在BlackHat 2014上神器Mimikatz的作者剖析了微软实现的Kerberos协议中存在的安全缺陷，并通过神器Mimikatz新添功能“Golden Ticket”展示了如何利用这些安全缺陷，让Kerberos把守的地狱之门为大家敞开。**

**一、快速介绍**

Kerberos 是Windows活动目录中使用的客户/服务器认证协议，为通信双方提供双向身份认证。相互认证或请求服务的实体被称为委托人（principal）。参与的中央服务器被称为密钥分发中心（简称KDC）。KDC有两个服务组成：身份验证服务（Authentication Server，简称AS）和票据授予服务（Ticket Granting Server，简称TGS）。在Windows域环境下，身份验证服务和票据授予服务可同时运行在任何可写域控服务器上。（PS:Kerberos其实是希腊神话中看守地狱大门的三头猛犬的名字，这只猛犬除了比一般的狗多出两个头外，还有喷火与喷硫酸的特异功能。）

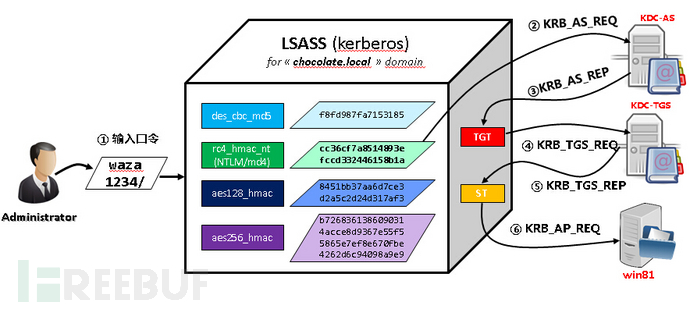
**1.相关术语**

委托人（principal）是一个具有唯一标识的实体，可以是一台计算机或一项服务，通过使用KDC颁发的票据来进行通信。委托人可以分为两类：用户和服务，分别具有不同种类的标识符。用户通过如“user@REALM”格式的用户主体名称（User Principal Name，简称UPN）来标识。记住REALM一定是大写的。例如用户“bob”在“bhusa.com”域中应该表示为bob@BHUSA.COM。

服务主体名称（Service Principal Name，简称SPN）是用于域中的服务和计算机账户。SPN的格式形如“serviceclass/host\_port/serviceName”。例如， 主机“dc1.bhusa.com”上LDAP服务的SPN可能类似于“ldap/dc1.bhusa.com”， “ldap/dc1”和“ldap/dc1.bhusa.com/bhusa.com”。参考全限定主机名和仅主机名，一个服务可能注册为多个SPN。

同通常是执行DNS查询来规范化主机名称。这就解释了DNS为什么是微软Kerberos环境中的一个必要组件。查询服务的“规范化”名称，然后生成请求服务的SPN。

**2.认证步骤**

[](http://image.3001.net/images/20141002/14122235056423.jpg)

Kerberos认证过程概览

用户要访问活动目录中的一项服务需经过以下几个步骤：

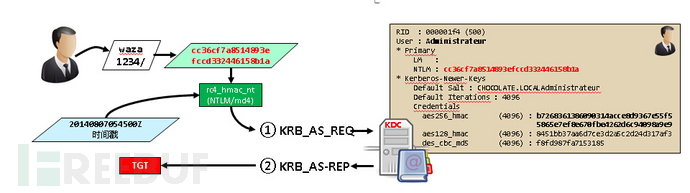
1.客户端对用户口令执行散列运算。此散列值（即用户密钥）成为客户端和KDC共享的长期密钥（**long term key**）。

2.**KRB\_AS\_REQ-**客户端加密一个时间戳，然后发送给身份验证服务。

3.**KRB\_AS\_REP-**身份验证服务会解密时间戳，若解密成功，表明了客户端获得某个特定用户的口令（即验证了用户的身份）。AS向客户端回复两条信息：

①短期会话密钥，用于客户端向KDC发起后续的请求，该消息经客户端的长期密钥加密。（此短期会话密钥仅适用于该客户端和KDC之间）

②票据授予票据（Ticket Granting Ticket，简称**TGT**），包含有关用户名、域名、时间和组成员资格等信息。该消息经仅可知KDC的密钥加密（在Windows环境中为krbtgt账户的NT-Hash）。**记住KDC不记录状态：客户端每次请求访问一项服务时，TGT都会被转发**。



客户端和Kerberos的预认证过程

4.**KRB\_TGS\_REQ**-客户端使用AS返回的会话密钥构建访问特定服务的请求。客户端把TGT连同请求一起发送到票据授予服务。

5.**KRB\_TGS\_REP**-票据授予服务解密TGT和服务请求，然后如果请求被允许，票据授予服务向客户端发送一个服务票据（Service Ticket，简称ST），包括两个部分：

①远程服务器的部分-包含请求用户的组成员资格、时间戳、用于客户端和远程服务器之间通信的会话密钥。使用远程服务器和KDC共享的长期密钥加密这部分消息。

②客户端的部分-包含用于客户端和远程服务器之间通信的会话密钥。使用步骤3中AS回复的短期会话密钥加密这部分消息。

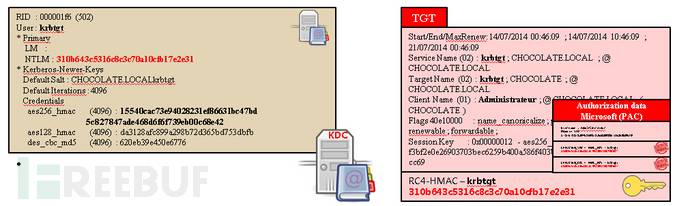
6.**KRB\_AP\_REQ**-客户端把服务票据中的服务器部分和请求一起发送到远程服务器。远程服务器将直接接受该服务器票据，并不需要和KDC直接通信，因为该票据是用远程服务器和KDC共享的长期密钥加密过的。解密成功即表明KDC已经允许了此次通信。

Kerberos信任模型的核心是每个委托人（principal）和KDC的通信是在利用仅双方可知的密钥构建的安全通道中进行。当委托人（principal）之间需要通信的时候，它们再使用KDC生成的会话密钥。

Kerberos也允许使用PKI和智能卡进行身份认证。用户会被提示输入一个智能卡的PIN码，而不是口令。Windows使用PIN码来访问智能卡上的公钥证书（public key certification）。利用智能卡的私钥签名该证书，并发送到KDC。KDC验证证书上的签名是否源于可信实体。然后KDC发送公钥证书加密过的TGT。既然信息只能被智能卡的私钥解密，用户也就通过了域的身份认证。然而，对于使用智能卡进行身份认证的账户来说，密码的散列值仍然存储在域控服务器上。此外，智能卡只能对“交互式会话（interactive sessions）”提供保护。也就意味着智能卡认证仅能用于登录域中的计算机。

**二、Kerberos信任完全依赖于KDC密码**

Kerberos协议是无状态的，因此密钥分发中心和票据授予服务并没记录以前的交互信息。因此票据授予服务所需使用的全部信息都位于TGT票据中。因为TGT使用KRBTGT的密码加密过，理论上讲网络上只有两方能够解密TGT：颁发票据的KDC和接受票据并创建访问网络资源的服务票据的票据授予服务。这种情况让KRBTGT成为系统中最重要的密码。最终结果是只要TGT被krbtgt账户密码正确地加密，TGT中的所有信息都是可信的。



**FreeBuf小科普**

krbtgt账户：每个域控制器都有一个“krbtgt”的用户账户，是KDC的服务账户，用来创建票据授予服务（TGS）加密的密钥。

**三、关于krbtgt账户和TGT的问题**

**1. 问题一：微软对长期密钥的散列值不加SALT**

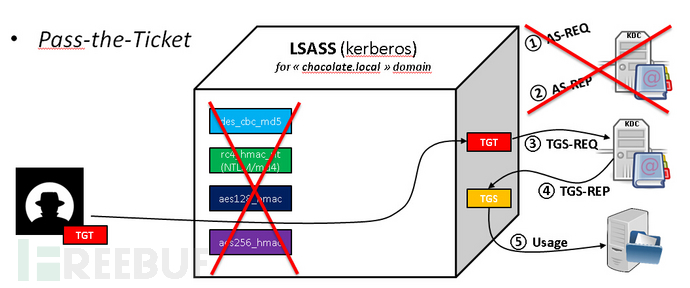
微软实现的Kerberos版本对MIT原始版本中的一个关键函数做了修改，最终降低了底层的安全性。在MIT原始版本中，首先在明文口令中添加字符串username@DOMAIN.COM，然后经过散列运算生成长期密钥。使用用户名给密码加盐，能够为碰巧密码相同的不同用户生成不同的散列值。只要涉及到密钥的操作，仍然需要使用实际的口令。

在微软实现版本中，口令转换成Unicode格式，再经过MD4运算生成没有加盐的密钥。这种情况也被称为NT-Hash，和微软十多年来存储长度大于14字符的密码的格式一样。缺少salt意味着任何需要密钥的操作能够直接地使用密码的散列版本，而不是使用实际的密码。这听上去像大家耳熟能详的“pass-the-hash”攻击的根本原因。

从一个攻击者的角度出发，如果能够提取该域的密码散列值，也就可以利用KRBTGT散列值来伪造TGT。虽然提取散列值看似难以实现，然而实际上，大部分渗透人员认为在普通的企业环境中这并不是一件困难的事情。我们会在白皮书下面的一个章节中作全面深入的解析。

**利用思路**

在微软活动目录中颁发的TGT是可移植的。由于Kerberos的无状态特性，TGT中并没有关于票据来源的标识信息。这意味着可以从某台计算机上导出一个有效的TGT，然后导入到该环境中其他的计算机上。新导入的票据可以用于域的身份认证，并拥有票据中指定用户的权限来访问网络资源。这种特别的攻击方法被称为“**pass-the-ticket**”攻击。



Pass-The-Tickets攻击思路

**2. 问题二：长期不变的密码**

krbtgt 密码的另一个问题是：它是整个活动目录中唯一的不会自动更新的密码。查看几个活跃网络中KRBTGT账户的年龄，账户已有五年或更多没更改密码的情况并不少见。在下面的例子中，此krbtgt密码是最后设置于2005年4月。



使用“net use”命令显示krgtgt的寿命

krbtgt密码仅在两种情况下将会自动更新。

**第一种情况**：域功能级别（domain functional level，简称DFL）从NT5（2000/2003）升级到NT6（2008/2012）。作为升级过程的一部分，KRBTGT账户的密码获得更改。每次域功能级别的升级都会面临一些特有的挑战，而这些挑战很大程度上是由网络其他部分上运行的软件版本所引发的。考虑升级过程可能耗费几个月甚至一年的时间，大部分公司不会轻易对现有网络进行升级操作。

**第二种情况**：利用域的恢复数据来实施域的裸机恢复（bare metal recovery）。域的恢复数据是在域初始运行时备份的。作为恢复过程的一部分，KRBTGT的密码需要改变两次。因为这是一个灾难恢复场景，并且需要域的恢复数据，而域的恢复数据很可能因为时间久远而被弄丢了。所以，该场景难以在现实世界中发生。

**FreeBuf小科普**

bare metal recovery (裸机恢复)：硬盘发生灾难性故障后对计算机进行完全恢复。该过程包括通过一个完全恢复点还原操作系统、文件系统、分区、卷和数据。

**四、客户端的有效性**

由于Kerberos是无状态的，TGT必须包含了同票据授予服务交互所需的全部信息。根据微软实现的Kerberos协议扩展（MS-KILE），KDC负责设置TGT中相应的标志位，而票据授予服务使用这些标志位来颁发服务票据。默认情况下，TGT携带了有关用户的组成员资格、票据所使用的加密类型以及票据的有效期等信息。此外，微软使用TGT来加强域上传统的和最新的身份认证策略。TGT包含一些传统的值来限制账户，例如登录时长、账户是否禁用、账户是否过期、密码失效的时间戳。

“本质上，客户端的有效性体现在TGT中的账户策略功能，因为客户端利用TGT向票据授予服务提出自己有什么安全限制。”

值得注意的是，Kerberos协议并没明确规定一个“票据授予服务如何验证账户是否被禁用”的机制，但是微软引入功能来让票据授予服务在颁发服务票据之前验证TGT是否被禁用。在后面的部分会围绕这个话题进行深入探讨。

**五、万能票据（Golden Ticket）**

综上所述，微软实现的Kerberos版本中存在的一系列问题所引发的后果是：Kerberos成为了攻击者实施Post-Exploitation的游乐场。如果攻击者能够攻陷KDC和提取KRBTGT散列值。然后利用这些有限信息，攻击者能够为委托人生成任意的TGT。Mimikatz工具把这种特性称之为“**万能票据**”（golden ticket，直译为“黄金票据”，为突出功能上的特点，意译译为“万能票据”）。万能票据拥有一些有趣的属性，会导致传统企业域中一些严重的安全问题。

首先，万能票据是全功能的TGT。也就意味着万能票据可用于Kerberos认证的任何服务。票据授予服务盲目地相信TGT中的信息，然后处理TGT并颁发服务票据。内存中插入万能票据并不需要提升权限。而且默认情况下，万能票据的有效期是10年。

其次，万能票据可以用来绕过当前Kerberos有关加密策略的要求。例如，可以使用DES或RC4加密算法创建一个TGT，即使该域明确支持AES，禁止使用DES或RC4。此情况会产生一个有趣的现象：TGT使用DES加密而服务票据使用AES加密。票据授予服务似乎并不担心TGT，也不拒绝异常行为，因为没有机制让票据授予服务报告关于策略的错误。

再次，万能票据并没启用任何高级账户策略的设置。微软添加了一个功能来验证服务票据的请求，以确保已禁用的TGT不能用于获得服务票据。然而，该功能的实现存在问题。只有当TGT的寿命超过20分钟时，票据授予服务才会验证TGT的有效性。如果TGT的寿命低于20分钟，票据授予服务将直接颁发服务票据，而不去验证TGT的有效性，默认情况下服务票据具有10小时的有效期。因为攻击者可以利用Mimikatz工具随心所欲的产生票据，所以攻击者只需清除旧的TGT，再替换为寿命少于20分钟的新票据，轻松突破20分钟的限制条件。

最终，万能票据可以被配置成任意用户和任意组的成员。这也可以创建一个票据，票据中任何用户都可以是任意组的成员。这可以用来绕过文件服务器或其他应用程序上基于用户组的访问限制。万能票据中的用户和SID不必在活动目录中真实存在。也就意味着可以为域中不存在的用户创建TGT,并仍然可以在TGT生命周期内前20分钟内从票据授予服务获得服务票据。

归根结底，在一个企业的活动目录的环境中，如果KRBTGT账户的丢失，也就意味着环境中的信任关系完全丧失。考虑到攻击者可以生成任意的Kerberos票据，能绕过对任意用户的账户策略，能够让用户成为任意组的成员，所以攻击者简直可以横行无阻。此外，鉴于KRBTGT账户极少更改散列值，攻击者可以在较长的时间内使用这些伪造的票据。

**六、攻击演示**

我们在此展示一些把NT-Hash升级到有关Kerberos票据的研究。假设攻击者能够获得一个特定用户名的NT-Hash，为什么不可以把NT-Hash转换为一个票据。

**1. 准备条件**

创建“万能票据”的所需条件：

①krbtgt账户的NT-Hash：该散列值是用于Kerberos的秘密密钥，仅位于域控服务器的活动目录中。所以攻击者必须攻陷域控服务器并提权至管理员权限；

②域账户名称：通常是域管理员“domain admin”；

③域名；

④域SID：可以从域用户的SID或通过sysinternal中psGetsid.exe获得；

**2.攻击演示**

在Windows中可以使用Mimikatz来实现。首先，我们把NT-HASH放入MSV1\_0，Kerberos服务账户的相关信息如下图所示：



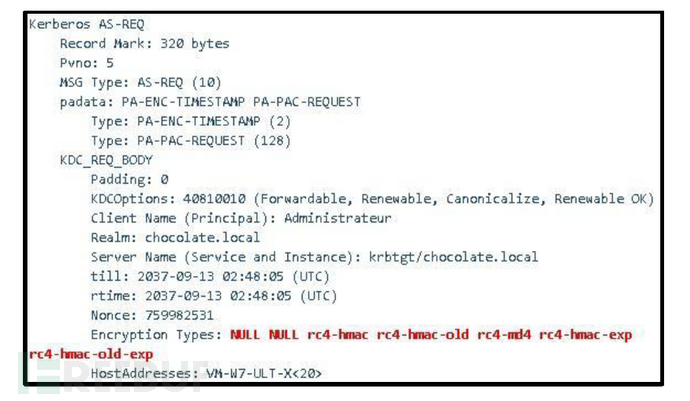
使用Mimikatz插入NT-Hash

当散列值插入到内存中，我们擦除掉所有其他的密钥，因为这些密钥可能干扰我们获得指定的票据。留下的密钥只有上图中插入到镜像的RC4密钥。



插入NT-Hash后内存中的加密密钥列表

然后我们尝试使用Kerberos访问一个服务。在此我们看到一个票据的请求。注意到支持的加密类型仅有RC4：



 发出的AS-REQ当我们尝试使用访问网络资源时

在此我们看到产生的TGT：



仅使用NT-Hash加密的TGT

这项技术在使用AES算法的Kerberos环境也是有效的。与预认证唯一的不同之处是使用AES密钥加密时间戳而不是传统的RC4（即NT-Hash）。

参考信息来源 Blackhat2014，本文综合了作者会上的文章和PPT的内容，各有取舍；同时也查阅了Kerberos相关资料，加上自己的理解组织成文，尽量保留了作者本意；本文涉及的技术内容较多，既有Kerberos网络协议，又有微软关于密码存储技术，还涉及到域控技术，我有些技术涉及不够深入，内容难免会出现一些错误，欢迎大家批评指正。后续会发布关于如何防范万能票据的文章，敬请期待！