

Anti-Forensic: Swipe Footprint with Timestamp

August 25, 2020 By Raj Chandel

In this article, we will learn how we can swipe our footprint after hacking the victim's system. We can achieve that with the help of the Timestamp feature provided by Metasploit Framework.

Let's understand the scenario

In this scenario, how a hacker can remove his footsteps in the victim system after the hack. So, that he won't get caught in the Forensic Investigation.

Objective: Learn to use the functionality of the Timestamp feature provided by the Metasploit Framework.

Ex: Changing the Date and Time of Modified, Created, Accessed of a particular file type.

Table of Content

- **Introduction to Timestamp**
- **Display MACE value**
- **Set the Modified date and time**
- **Set the Accessed date and time**
- **Set the Created date and time**
- **Set the Entry Modified date and time**
- **Set All four attributes at once**
- **Set the MACE attributes equal to supplied file**

Introduction to Timestamp

As we all know with file systems is be like walking in the snow.... we will leave footprints. It will depend on how detailed those footprints are, how much we can learn from them, and how long we last all depends on various circumstances.


The art of analyzing these artifacts is known as Digital Forensics. For various explanations, when conducting a penetration test, we may want to make it tough for a forensic analyst to determine the movements that we took.

To avoid detection by the **forensic investigation is simple: Don't touch the filesystem!** This is the beauty of Meterpreter, it will load into memory without writing anything to disk, greatly minimizing the pieces it leaves on a system.

But in Some cases, we may have to interact with the filesystem in some way. In these cases, Timestamp will be a great tool.

To know the full functionality of the Timestamp feature we just need to take a meterpreter session and follow these commands.

```
timestamp help
```


```
meterpreter > timestamp help   
Usage: timestamp <file(s)> OPTIONS  
OPTIONS:  
-a <opt> Set the "last accessed" time of the file  
-b       Set the MACE timestamps so that EnCase shows blanks  
-c <opt> Set the "creation" time of the file  
-e <opt> Set the "mft entry modified" time of the file  
-f <opt> Set the MACE of attributes equal to the supplied file  
-h       Help banner  
-m <opt> Set the "last written" time of the file  
-r       Set the MACE timestamps recursively on a directory  
-v       Display the UTC MACE values of the file  
-z <opt> Set all four attributes (MACE) of the file
```

Display MACE value

This feature helps us to Display MACE values, where MACE stands **M**odified **A**ccessed **C**reated **E**ntry **M**odified. Through this feature, we can see these values and see if these values are modified during the hack or not.

To view, these details follow this command.

```
timestamp note.txt -v
```

```
meterpreter > timestamp note.txt -v   
[*] Showing MACE attributes for note.txt  
Modified      : 2011-08-07 09:46:08 -0400  
Accessed      : 2011-08-07 09:46:08 -0400  
Created       : 2011-08-07 09:46:08 -0400  
Entry Modified: 2011-08-07 09:46:08 -0400
```

Change Modified date and time

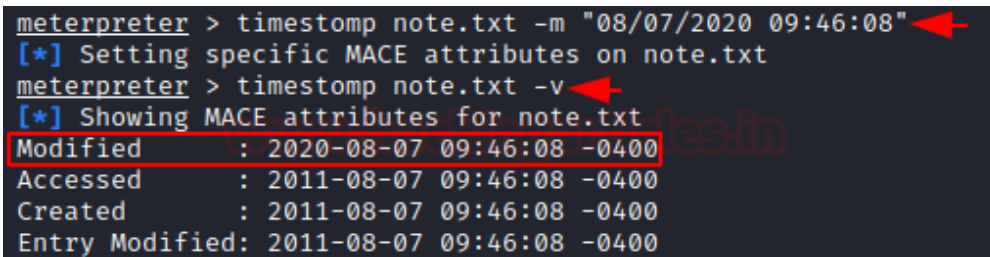
It is the date and time when a new version of the file, which is created better known as the last activity. Where last activity is the **date and time** when changes are made to the item's metadata.

Now we can change this modified date and time as per our need, with the help of **[-m]** parameter. By this command.

```
timestamp note.txt -m "08/07/2020 09:46:08"
```

we can also check that whether we can able to modify the date and time or not, with the help of [-v] parameter. As we can check the below screenshot, we have successfully able to modify the date and time.

```
timestamp note.txt -v
```



```
meterpreter > timestamp note.txt -m "08/07/2020 09:46:08"
[*] Setting specific MACE attributes on note.txt
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified       : 2020-08-07 09:46:08 -0400
Accessed      : 2011-08-07 09:46:08 -0400
Created       : 2011-08-07 09:46:08 -0400
Entry Modified: 2011-08-07 09:46:08 -0400
```

Change Accessed date and time

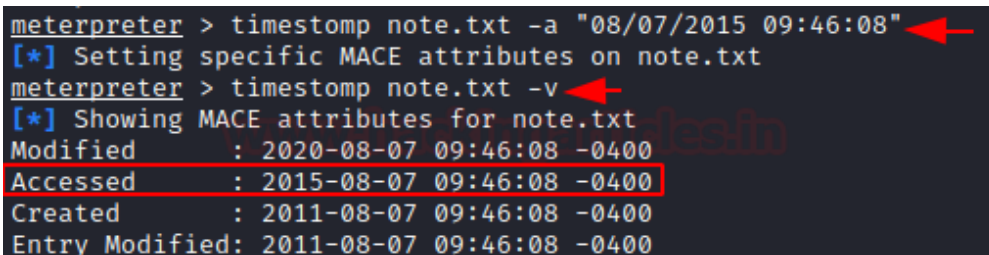
It is the date and time we accessed the material or file. It can be listed as day, month, and the year it is also included at the end of the citation.

Now we can change this accessed date and time as per our need, with the help of [-a] parameter. By this command.

```
timestamp note.txt -a "08/07/2015 09:46:08"
```

we can also check that whether we can able to change the date and time or not, with the help of [-v] parameter. As we can check the below screenshot, we have successfully able to change the date and time.

```
timestamp note.txt -v
```



```
meterpreter > timestamp note.txt -a "08/07/2015 09:46:08"
[*] Setting specific MACE attributes on note.txt
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified       : 2020-08-07 09:46:08 -0400
Accessed      : 2015-08-07 09:46:08 -0400
Created       : 2011-08-07 09:46:08 -0400
Entry Modified: 2011-08-07 09:46:08 -0400
```

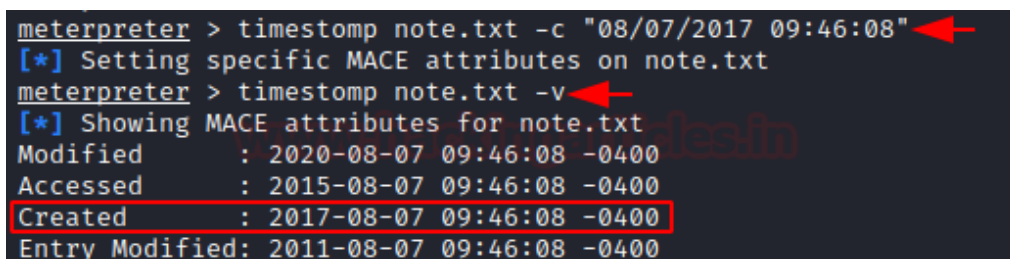
Change Created date and time

The created date is recorded when the file was created. Now we can change this created date and time as per our need, with the help of [-c] parameter. By this command.

```
timestamp note.txt -c "08/07/2017 09:46:08"
```

we can also check that whether we can able to change the date and time or not, with the help of [-v] parameter. we can check the below screenshot we have successfully able to change the date and time.

```
timestamp note.txt -v
```



```
meterpreter > timestamp note.txt -c "08/07/2017 09:46:08"
[*] Setting specific MACE attributes on note.txt
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified      : 2020-08-07 09:46:08 -0400
Accessed      : 2015-08-07 09:46:08 -0400
Created       : 2017-08-07 09:46:08 -0400
Entry Modified: 2011-08-07 09:46:08 -0400
```

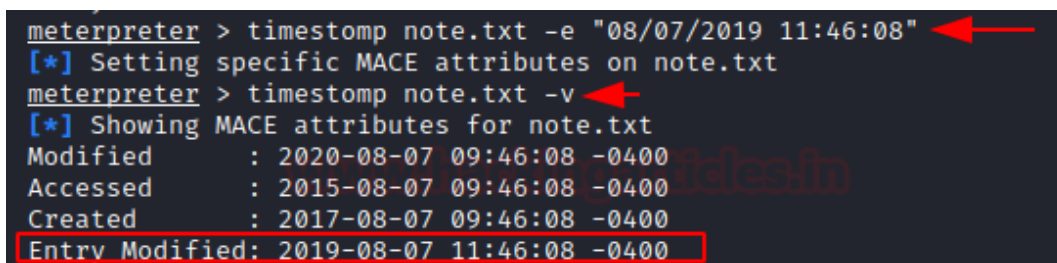
Change of Entry Modified date and time

It is the date and time of the last entry modified in the particular file or material. Now we can change this entry modified date and time as per our need, with the help of [-e] parameter. With the help of this command.

```
timestamp note.txt -e "08/07/2019 09:46:08"
```

we can also check that whether we can able to change the date and time or not, with the help of [-v] parameter. we can check the below screenshot we have successfully able to change the date and time.

```
timestamp note.txt -v
```



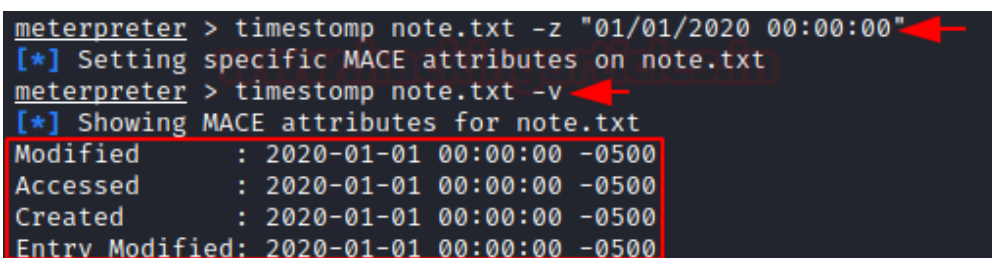
```
meterpreter > timestamp note.txt -e "08/07/2019 11:46:08"
[*] Setting specific MACE attributes on note.txt
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified      : 2020-08-07 09:46:08 -0400
Accessed      : 2015-08-07 09:46:08 -0400
Created       : 2017-08-07 09:46:08 -0400
Entry Modified: 2019-08-07 11:46:08 -0400
```

Set All four attributes at once

If we want to set all four MACE attributes [Modified, Accessed, Created, and Entry Modified]. It will change the whole investigation scenario, easy to us swipe the footprints of the hack.

Now we can change all MACE attributes date and time as per our need, with the help of [-z] parameter. With the help of this command.

```
timestamp note.txt -z "01/01/2020 00:00:00"
```



```
meterpreter > timestamp note.txt -z "01/01/2020 00:00:00"
[*] Setting specific MACE attributes on note.txt
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified       : 2020-01-01 00:00:00 -0500
Accessed       : 2020-01-01 00:00:00 -0500
Created        : 2020-01-01 00:00:00 -0500
Entry Modified: 2020-01-01 00:00:00 -0500
```

we can also check that whether we can able to change these dates and times or not, with the help of [-v] parameter. we can check the below screenshot we have successfully able to change these dates and times.

```
timestamp note.txt -v
```

Set the MACE attributes equal to supplied file

This amazing feature helps us to give the MACE attributes value equal to that particular supplied file. The first file which we use in this practical is **file.txt**. We use the [-v] parameter to display its MACE attributes values.

```
timestamp file.txt -v
```

Now, we have cross-checked the MACE attributes values of the **note.txt**, with the help [-v] parameter.

```
timestamp note.txt -v
```

After this, we use the [-f] parameter to change the MACE values of file.txt with note.txt. To achieve that we give the proper path of the note.txt file. Like this

```
timestamp file.txt -f "C:\Users\SKS19\Downloads\note.txt"
```

Now, we can also check that whether we can able to change these dates and times or not, with the help of [-v] parameter. we can check the below screenshot we have successfully able to change these dates and times. Through this feature, we can sweep our footprints during the hack.

```
timestamp file.txt -v
```

```
meterpreter > timestamp file.txt -v
[*] Showing MACE attributes for file.txt
Modified      : 2020-08-07 10:21:55 -0400
Accessed      : 2020-08-07 10:21:55 -0400
Created       : 2020-01-01 00:00:00 -0500
Entry Modified: 2020-08-07 10:21:58 -0400
meterpreter > timestamp note.txt -v
[*] Showing MACE attributes for note.txt
Modified      : 2020-01-01 00:00:00 -0500
Accessed      : 2020-01-01 00:00:00 -0500
Created       : 2020-01-01 00:00:00 -0500
Entry Modified: 2020-01-01 00:00:00 -0500
meterpreter > timestamp file.txt -f "C:\Users\SKS19\Downloads\note.txt"
[*] Pulling MACE attributes from C:\Users\SKS19\Downloads\note.txt
[*] Setting specific MACE attributes on file.txt
meterpreter > timestamp file.txt -v
[*] Showing MACE attributes for file.txt
Modified      : 2020-01-01 00:00:00 -0500
Accessed      : 2020-01-01 00:00:00 -0500
Created       : 2020-01-01 00:00:00 -0500
Entry Modified: 2020-01-01 00:00:00 -0500
```

Conclusion: We have learned, how a hacker can remove his footsteps in the victim system after the hack. So, that it won't get caught in the Forensic Investigation.

Although there are many different foundations of timeline information on the Windows system Other than just MACE times. If a forensic investigator came across a system that had been adapted in this manner, they would be successive to these substitute information sources.