Project-2

Vineeth Mylavarapu

ID.801337050

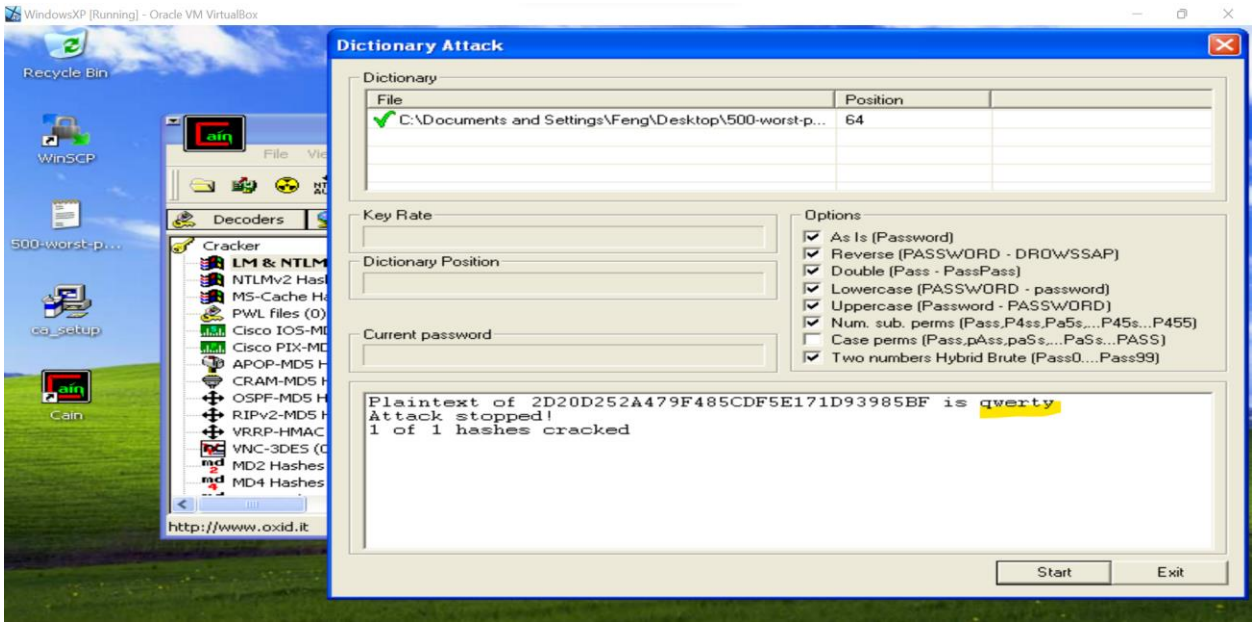
College of Computing and Informatics

MS in Cybersecurity

Task-1:

1. Password discovered from Task 1

A. As Instructed, I have tried cracking password of User **test1** with Cain Application. I have used dictionary attack with help of file 500-worst passwords. Please find the below screenshot for your reference. In the below screenshot, I have chosen **qwerty** which is very easy to crack.



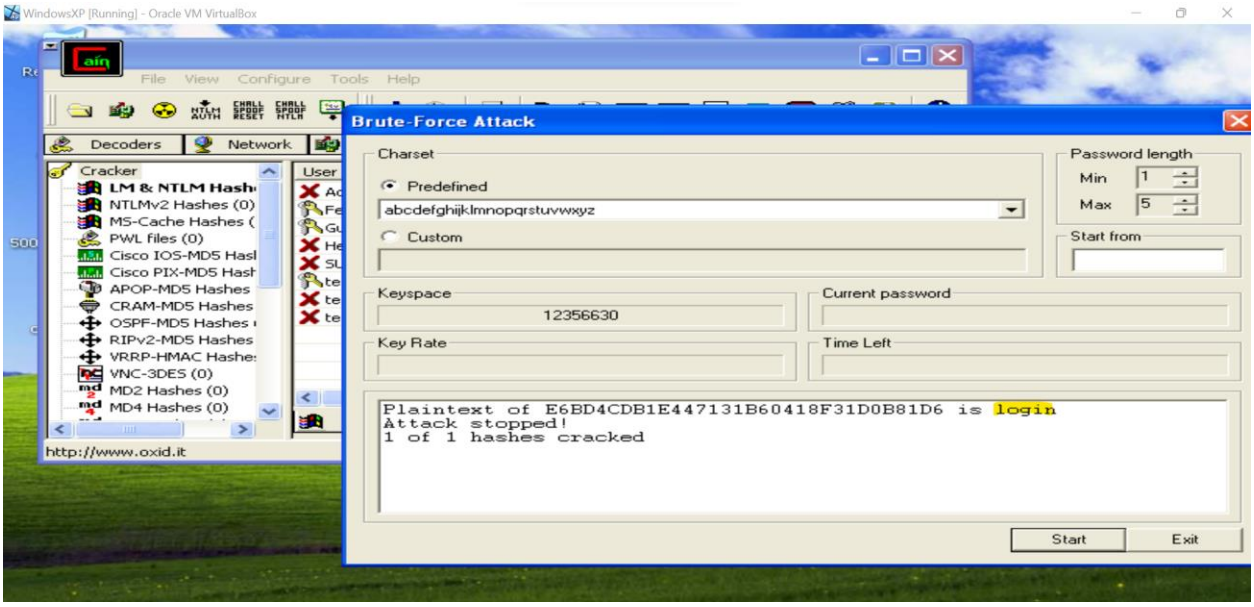
Task-2:

Brute Force Attack Table

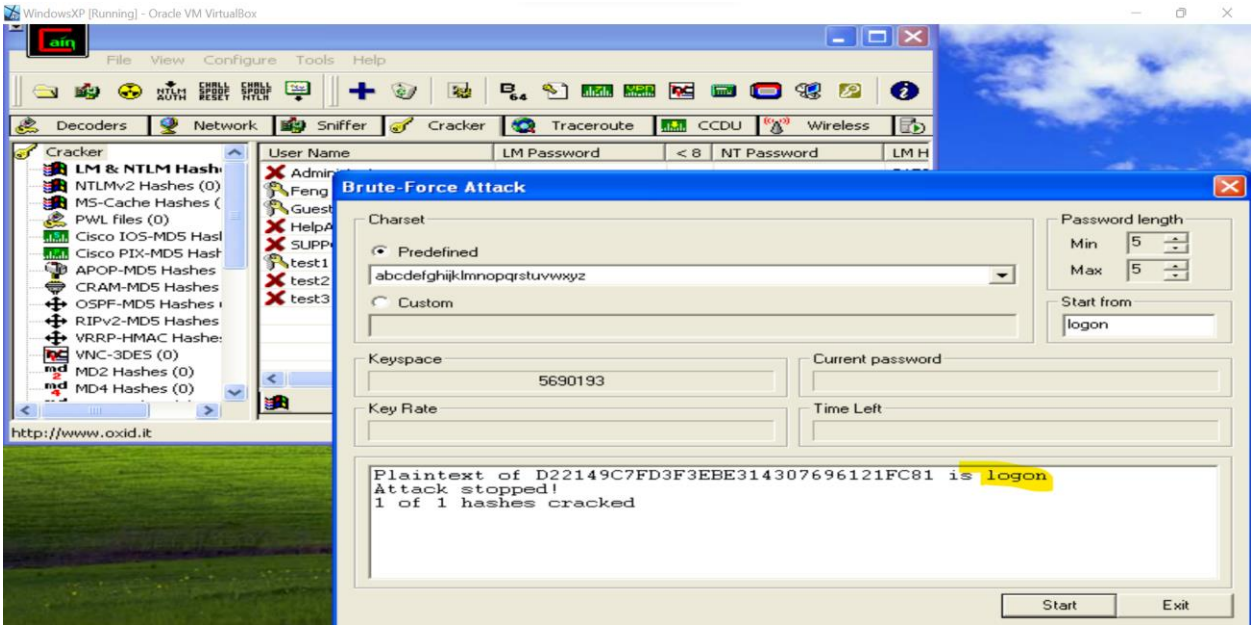
	Password Description	Chosen Password	Charset	Time Taken
1	Lowercase letters only (length 5)	Test1- **login** Test2- **logon** Test3- **chips**	abcdefghijklmnopqrstuvwxyz	Test1= >1sec Test2= >1sec Test3= >1sec
2	Lowercase, uppercase letters and numbers from 0-9 (length 5)	Test1- **PisP1** Test2- **Ccd23** Test3- **MaLa4**	abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789	Test1= 30secs Test2= 53secs Test3= 52secs
3	Lowercase, uppercase letters, numbers from 0-9 and symbols (length 5)	Test1- **P3p$!** Test2- **R&$h4** Test3- **T0d&Y**	abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789!@#$%^&*	Test1= 4min 54s Test2= 4min 23s Test3= 4min 0s

- **<u>Scenario-1: Lowercase Letters only (Length 5)</u>**

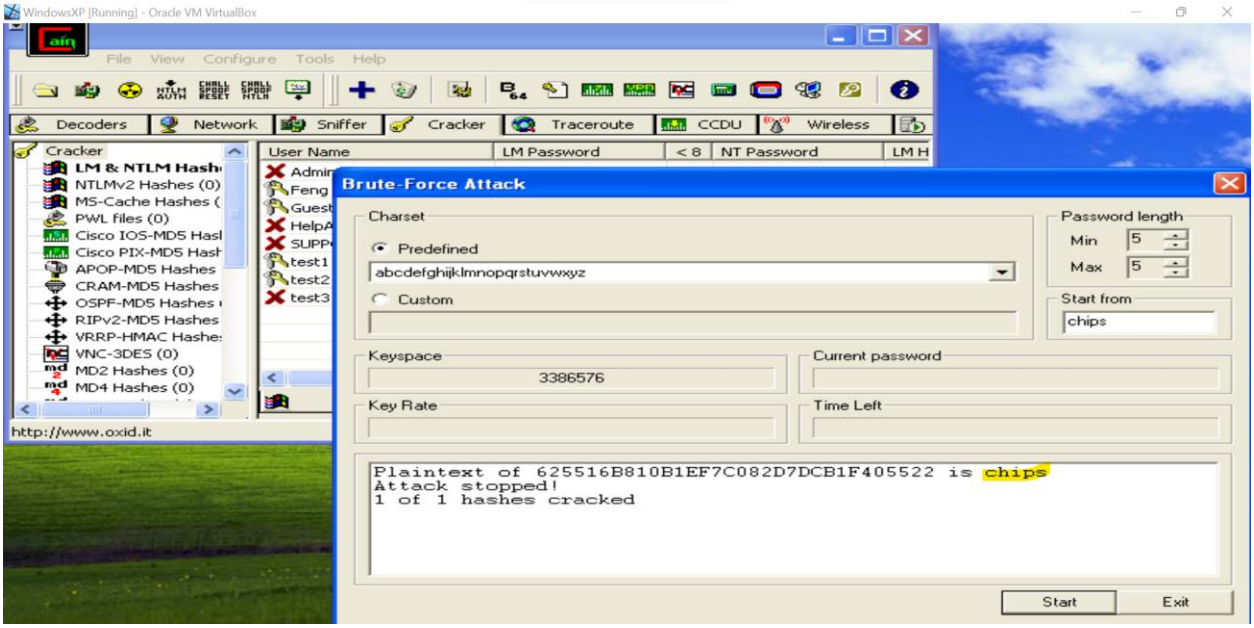
User Test1: I have created a password for test1 as **login** and using brute force I have cracked it less than 1sec. PFB Screenshot and above details for your reference.



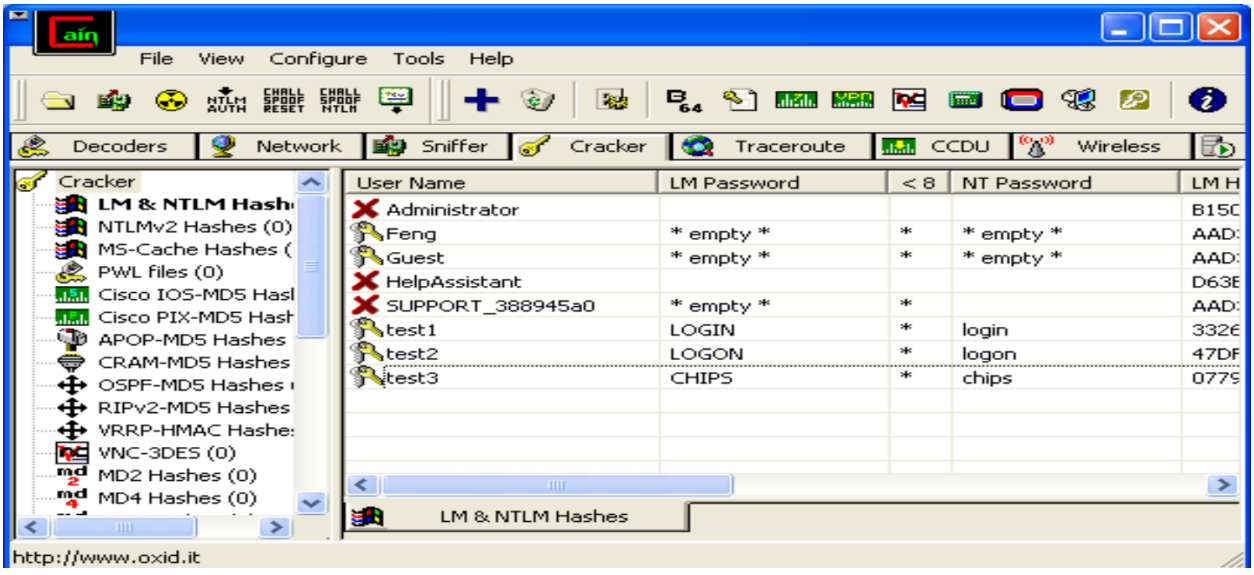
User Test2: I have created a password for test2 as **logon** and using brute force I have cracked it less than 1sec. PFB Screenshot and above details for your reference.



User Test3: I have created a password for test3 as **chips** and using brute force I have cracked it less than 1sec. PFB Screenshot and above details for your reference.

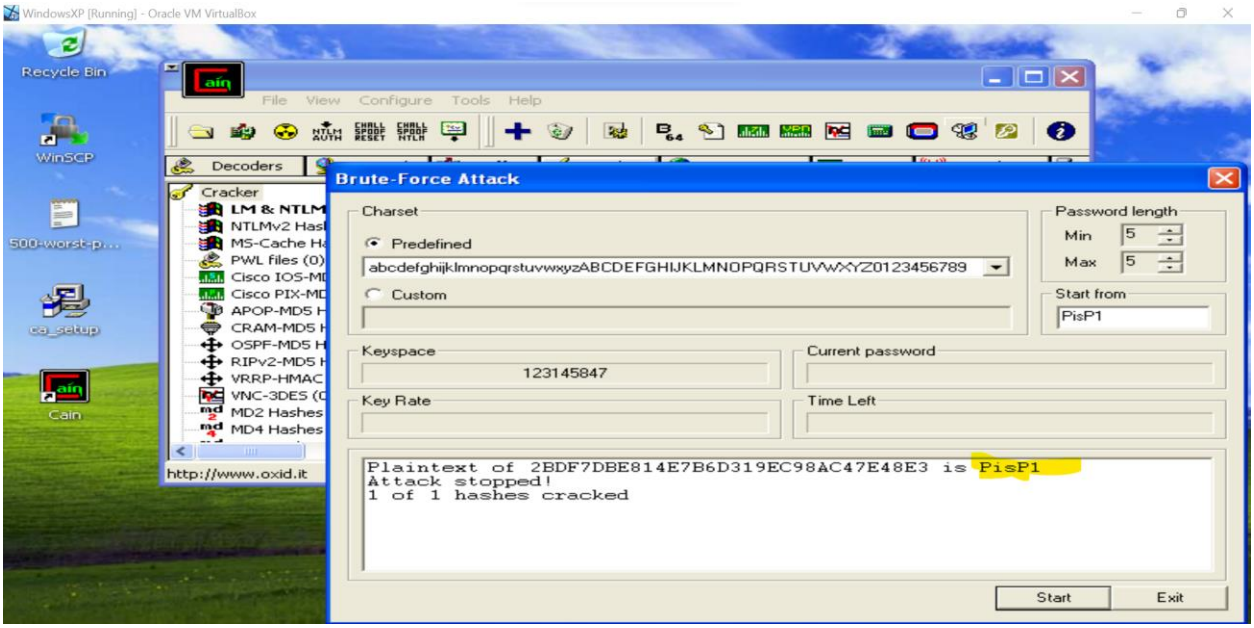


Please find the below screenshot from ca application which shows all the cracked passwords for all users.

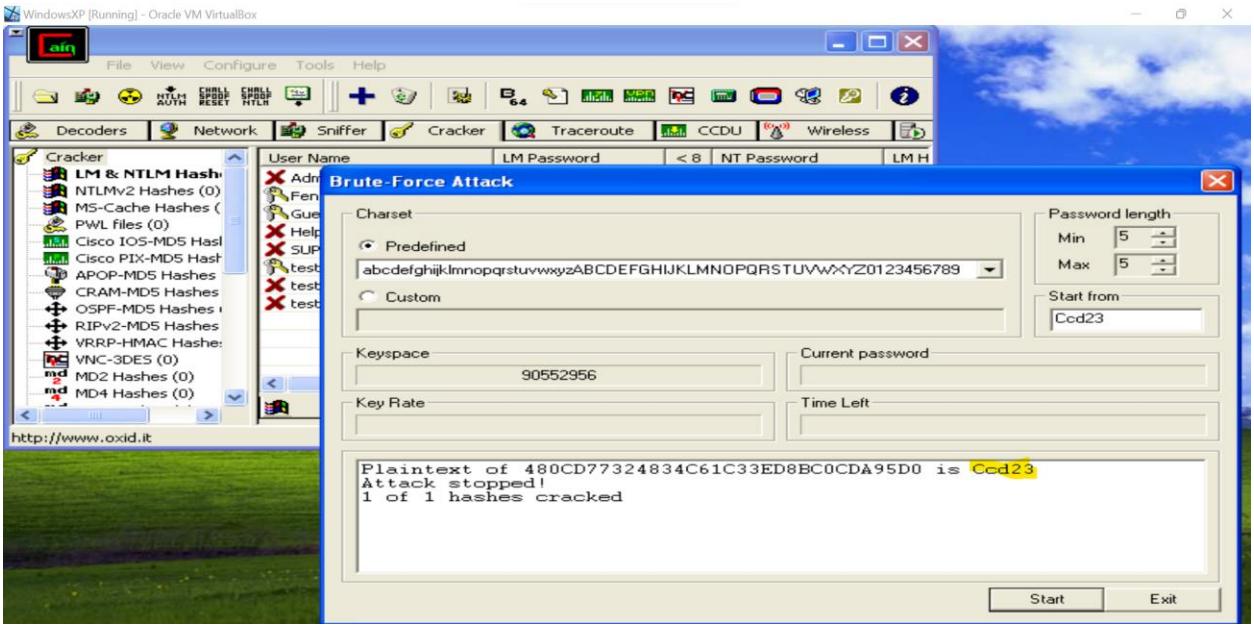


- **<u>Scenario-2: Lowercase, Uppercase letters and numbers from 0-9 (Length 5)</u>**

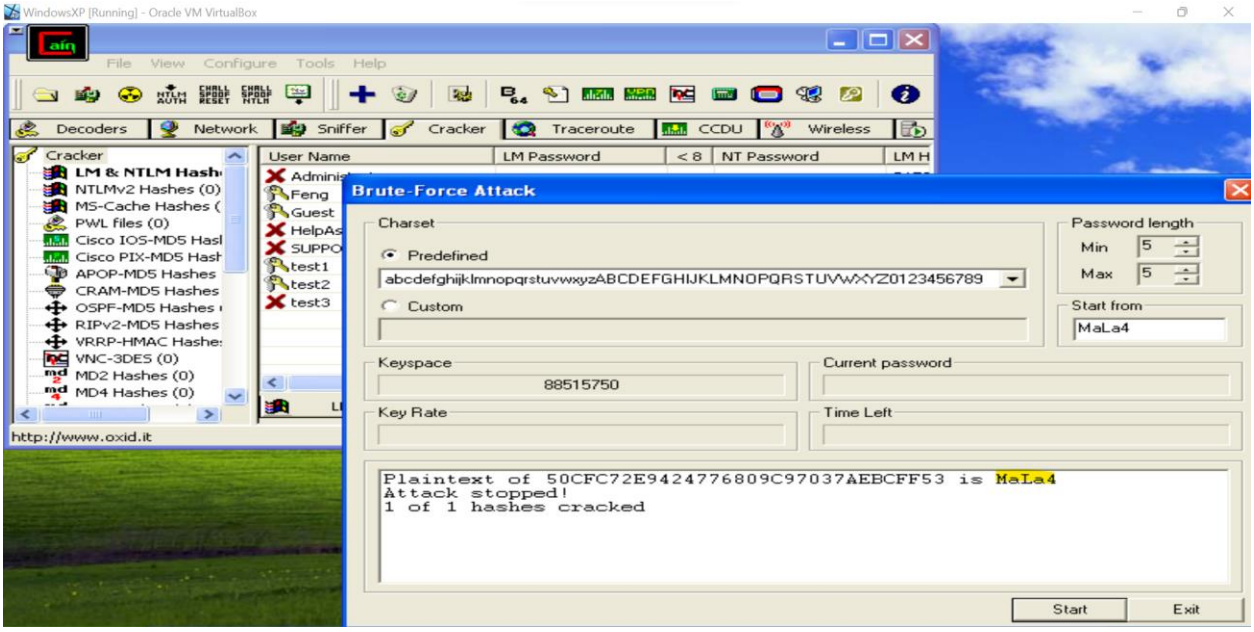
User Test1: I have created a password for test1 as **PisP1** and using brute force I have cracked it in 30secs. PFB Screenshot and above details for your reference.



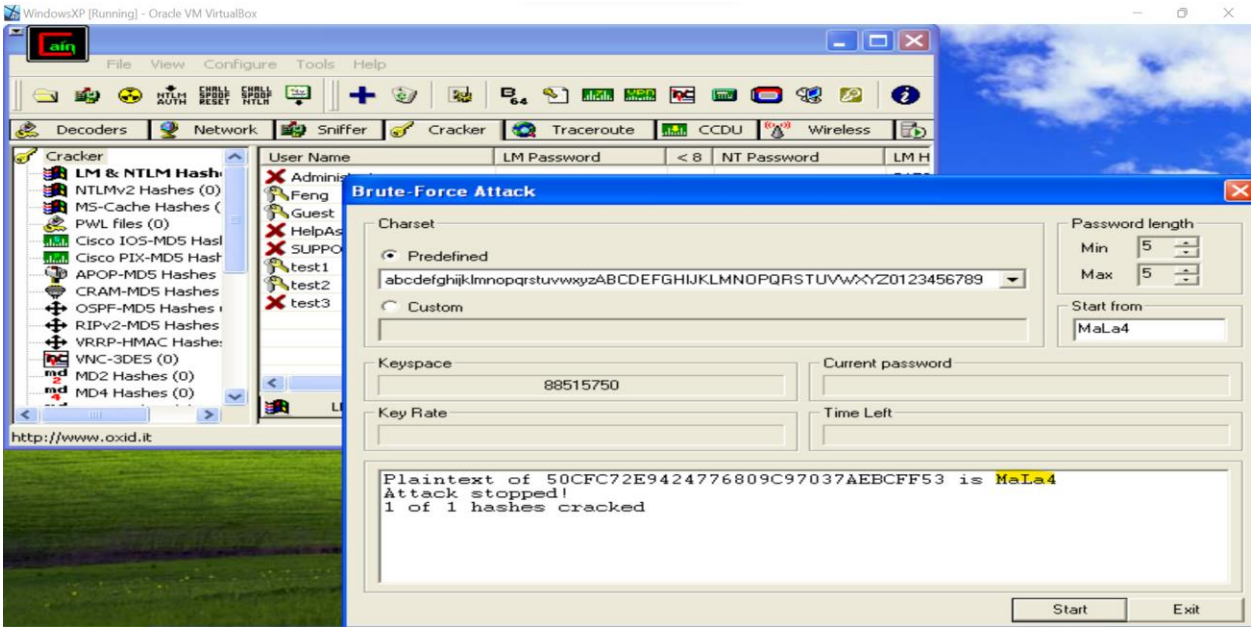
User Test2: I have created a password for test2 as **Ccd23** and using brute force I have cracked it in 53secs. PFB Screenshot and above details for your reference.



User Test3: I have created a password for test3 as **MaLa4** and using brute force I have cracked it in 53secs. PFB Screenshot and above details for your reference.

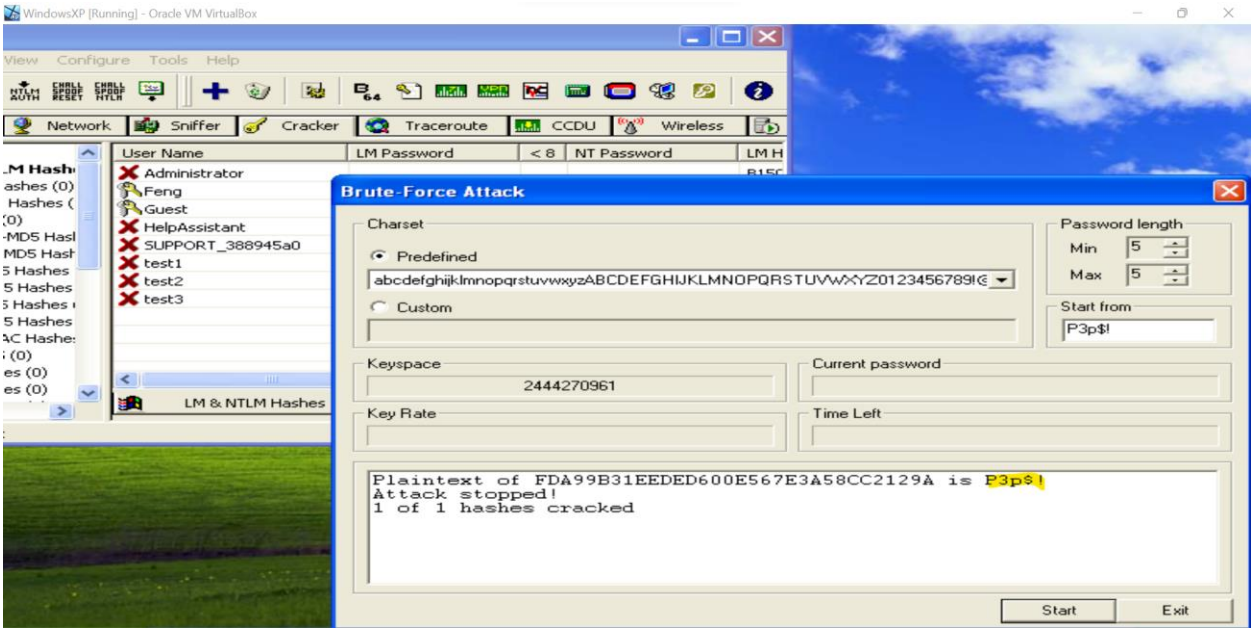


Please find the below screenshot from ca application which shows all the cracked passwords for all users.

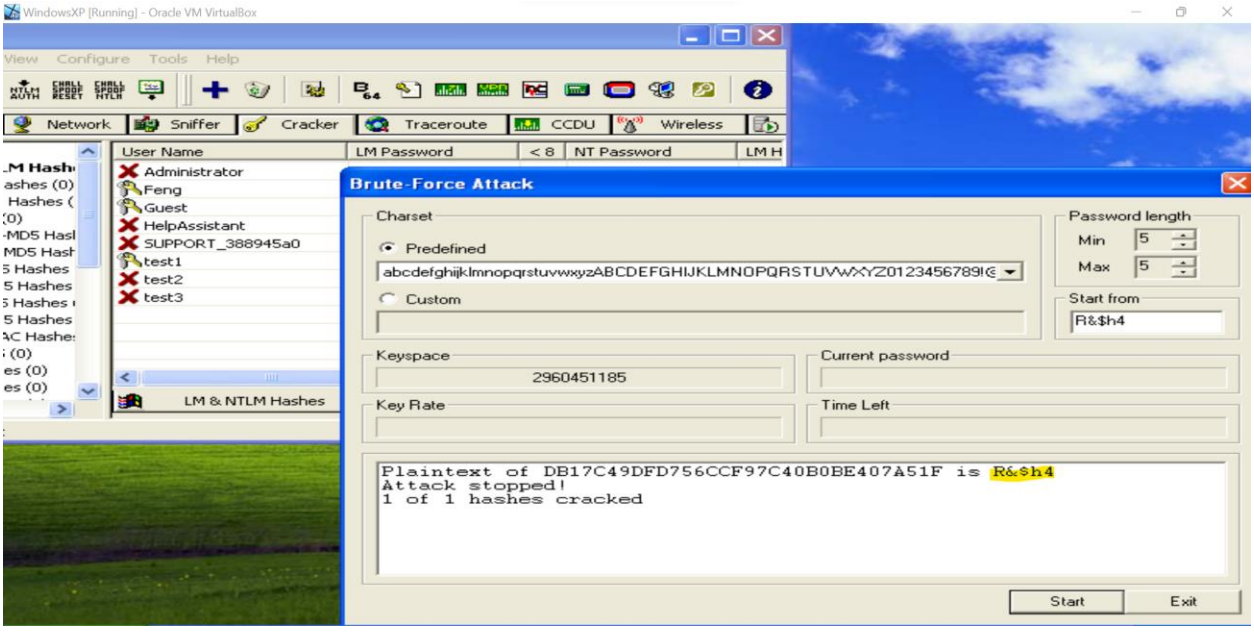


- **<u>Scenario-3: Lowercase, Uppercase letters, numbers from 0-9 and symbols (Length 5)</u>**

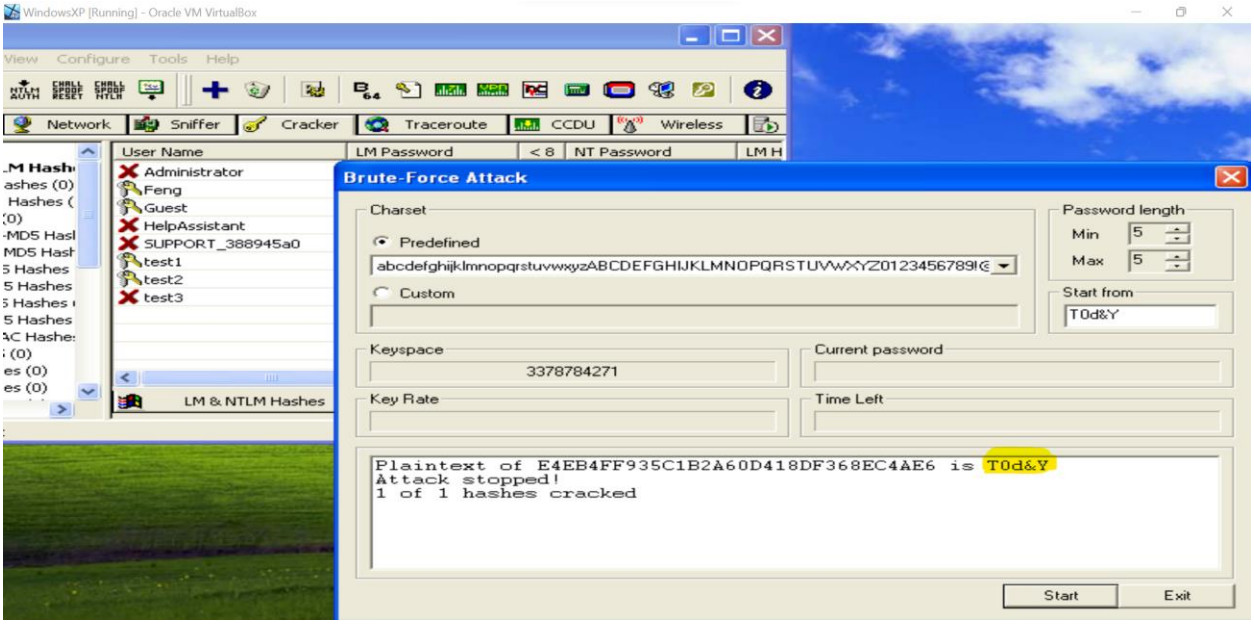
User Test1: I have created a password for test1 as **P3p$1** and using brute force I have cracked it in 4min 54secs. PFB Screenshot and above details for your reference.



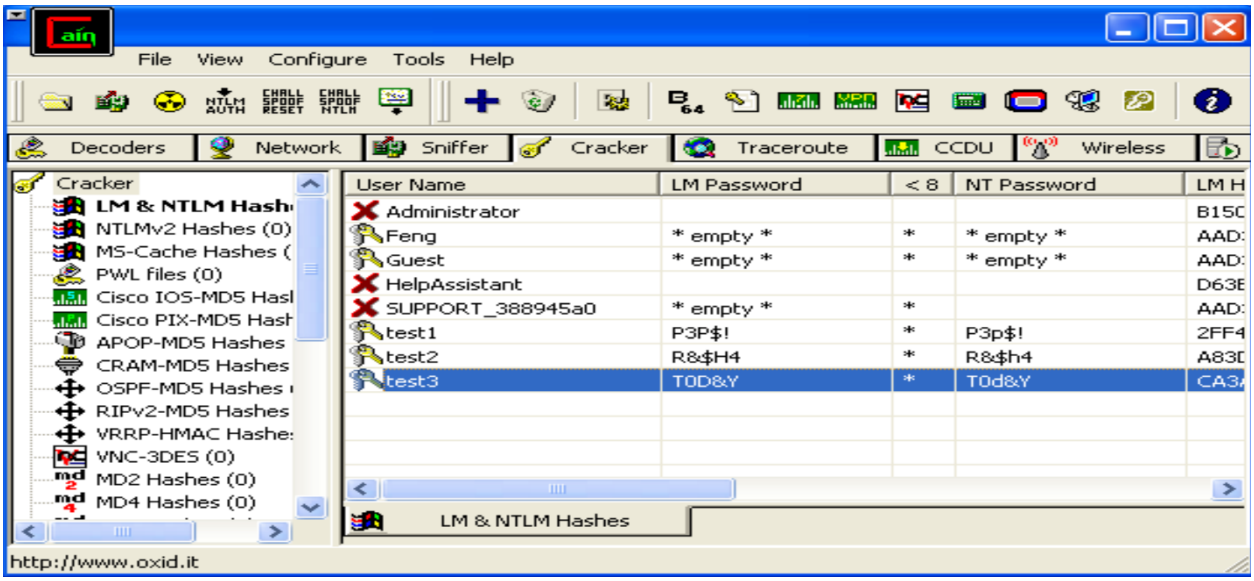
User Test2: I have created a password for test2 as **R&$h4** and using brute force I have cracked it in 4min 29secs. PFB Screenshot and above details for your reference.



User Test3: I have created a password for test3 as **T0d&Y** and using brute force I have cracked it in 4mins. PFB Screenshot and above details for your reference.



Please find the below screenshot from ca application which shows all the cracked passwords for all users.



- Answer the question: When you created passwords for the brute force attack, would Cain & Abel have finished faster if your password didn't include all the character types in the password description? So, for example if the description said "lower and uppercase letters", and if your chosen password was "aaa", would Cain and Abel have discovered it faster than if you had chosen "aBC"? Remember that in real scenarios, if you were trying to recover a password using a tool like Cain & Abel, you would not know what the password was, only what the password space was!

    A. The Cain & Abel application estimates that scenario 1 have taken 1 second to complete, scenario 2 took nearly 1 minute, and scenario 3 with 4 minutes and around. The results in scenario 3 shows that cracking passwords with a more complex character set takes longer. The application can attempt up to 17 million passwords per second, which is a substantial amount which had multiple permutations and computations.

    The time it takes for the Cain & Abel application to crack a password using a brute force algorithm is largely dependent on the character set used and the length of the password. Shorter passwords with a limited character set can be cracked quickly, while longer passwords with a larger character set will take longer time to crack. This is because the number of possible combinations increases as the length of the password increases. It is recommended to use a long password with a wide range of characters, including lowercase letters, uppercase letters, numbers, and symbols, for better security. Maintaining a hexadecimal type of passwords are difficult to crack.

    Cracking passwords of the same length and character set, such as "aaa" and "aBC", will take roughly the same amount of time, regardless of the size of the character set. This was confirmed through a personal test on my machine, where both passwords were cracked in less than a second.