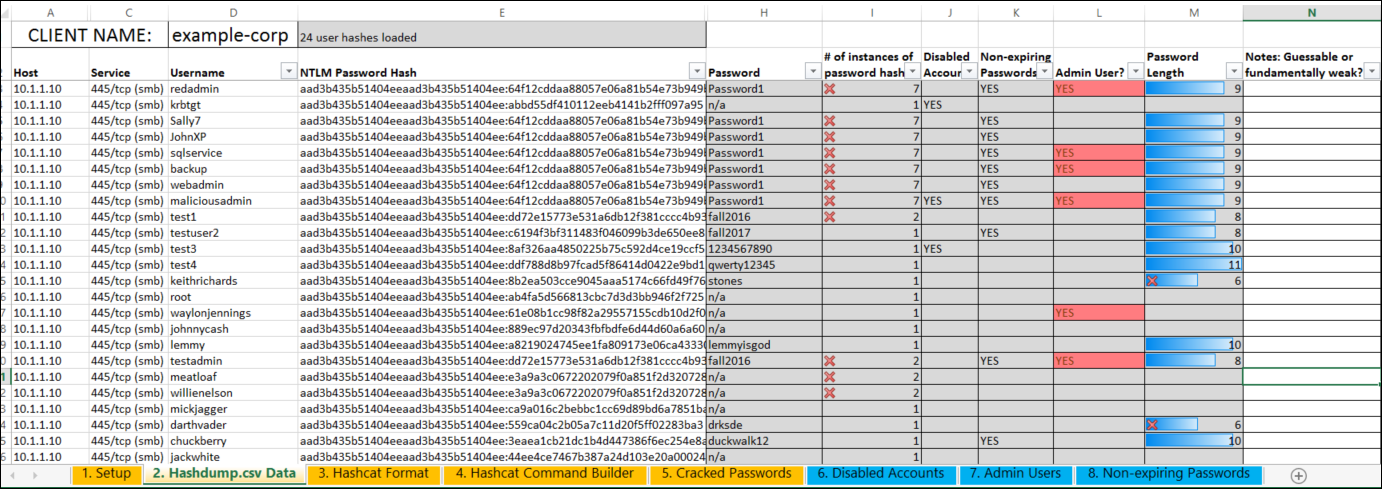
Hashdump Analysis tool

In the past I found that it was very difficult, overly complicated, and error prone to analyze dumped domain hashes beyond just identifying cracked passwords-- especially in very large environments. Attached is the latest version of a tool that I made to use on my own pentests to more efficiently crack and analyze domain user password hashes that I’ve dumped from compromised domain controllers. It organizes the dumped hash data in a spreadsheet and uses a bunch of formulas to extract, compare, analyze, count, and parse information about the hashes and their corresponding passwords (if we succeed in cracking the hashes). The tool also streamlines and organizes the hashcat workflow.

The hashdump analysis tool quickly organizes the data and displays:

* + Which accounts are sharing passwords
  + How many separate instances of password reuse are occurring, including instances when we cannot crack the password but can observe that the hashes are identical between multiple accounts
  + If passwords are being shared across privilege boundaries (i.e. a user and an admin share the same password)
  + Password length and flags anything less than the PCI standard of 7 characters
  + Which accounts have non-expiring passwords
  + Which accounts are domain admins
  + Which accounts are disabled (so they can be sorted out)

All of the data can be sorted on multiple levels (via drop down menus) to drill down and identify trends, which translate into insights about the client environment, which means delivering more value to clients. The hashdump analysis tool also formats the dumped hashes into hashcat format, builds the hashcat command for convenient copy/paste into the cracking server, and parses the hashcat output from the cracking server.



Overview of the hashdump analysis spreadsheet

The hashdump analysis spreadsheet has 8 numbered and color coded tabs. The tabs are intended to guide you through the tool and to be filled out in order. The color coding is used to indicate which tabs/cells are required to be filled out, which tabs/cells are optional (but recommended), and which cells are calculated by formulas and therefore shouldn’t be edited. Once all of the tabs are filled out, the 2nd tab becomes the main workspace and all data analysis happens there. I wrote up a brief tutorial below to walk you through how to use the tool and have included a completed example. All of the data in the screenshots and in the example spreadsheet are from my test lab.

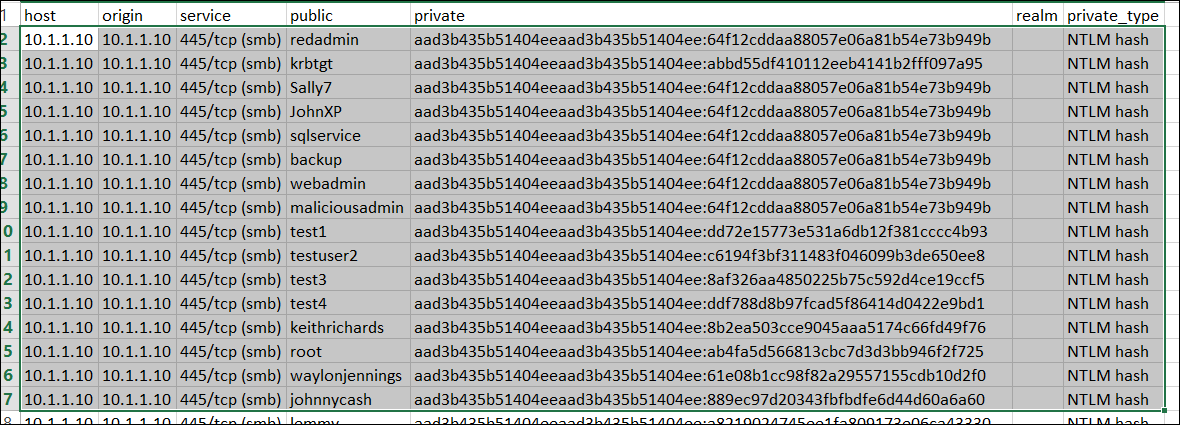
Tutorial

* + Get a Meterpreter shell on a domain controller using your preferred method and escalate to SYSTEM if you don’t already have SYSTEM privs. Afterwards, use the smart\_hashdump post exploit module in Metasploit to dump the domain password hash database. The module will write the data to a file and to the screen, but we don’t want to use that- using smart\_hashdump will automatically store the hashes in Metasploit’s credential database which will make the data easier to work with. **Using the hashdump tool that’s built into Meterpreter won’t do this, so use smart\_hashdump**. The spreadsheet formatting is set up to work with smart\_hashdump output, but this tool will work with other tools as well. See step 4 below.

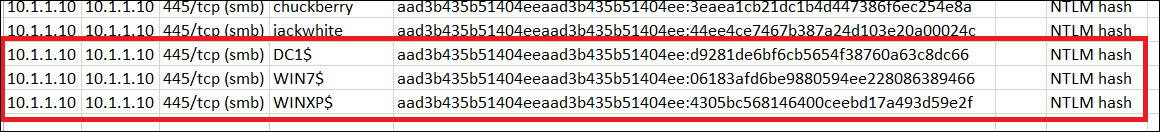
* + After extracting the domain password hash database with smart\_hashdump, write the creds to a .csv



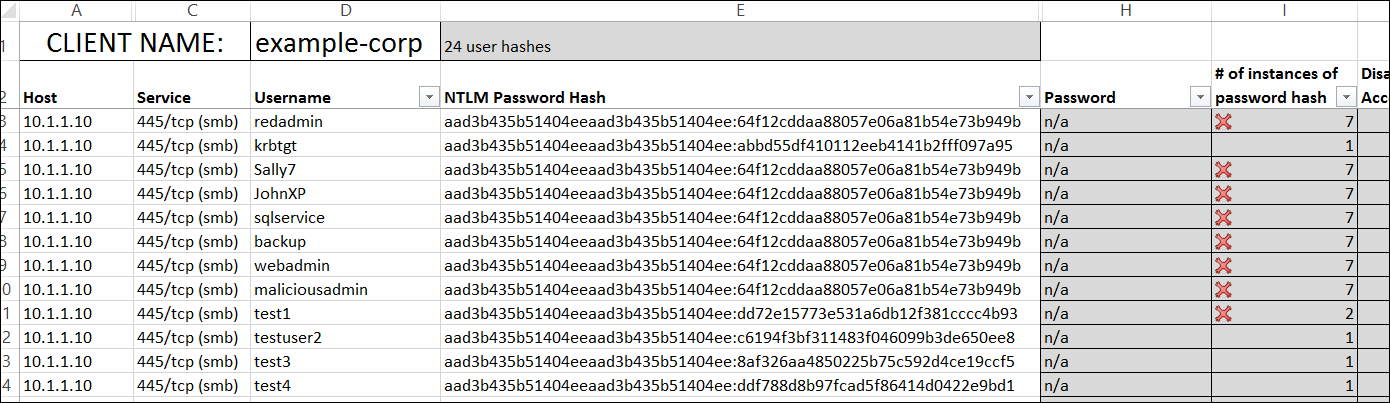
* 1. Open the .csv in Excel and copy *all* of the data for the **user** accounts.



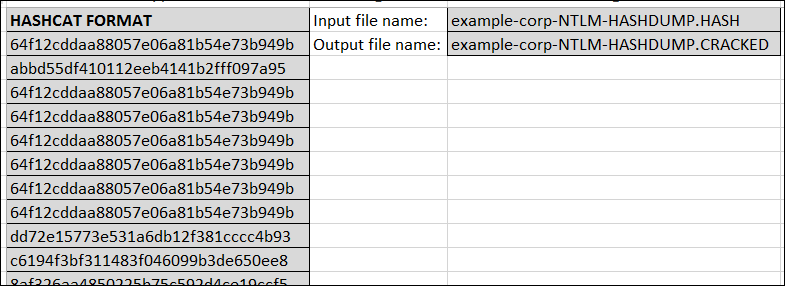
Don’t copy the machine account hashes- these are not considered crackable and should be discarded. These accounts can be easily identified because they end with a “$” and will appear at the end of the .csv file.



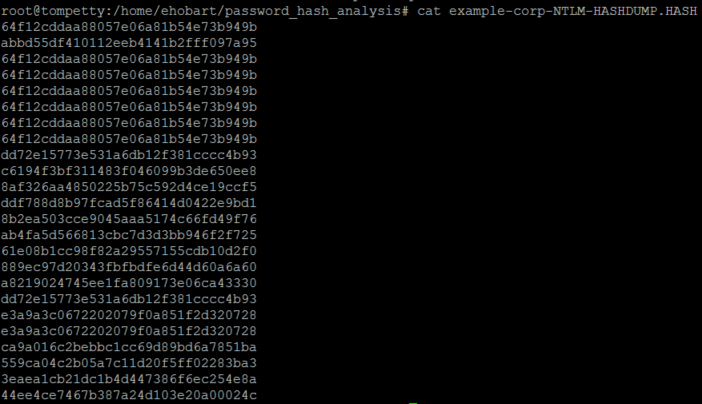
* 1. Open the hashdump analysis spreadsheet and fill out the client name on first tab (“1. Setup”). Paste in the user hash data into the second tab (“2. Hashdump.csv Data”). The hashdump analysis spreadsheet is set up to automatically hide some of the columns from the .csv that are not relevant. ) If you’re working with the output of a tool other than smart\_hashdump in Metasploit, that’s no problem. Just copy the usernames into Column D and ntlm hashes into Column E.

Here you can see that without even doing any password cracking, we’ve already identified instances of passwords being shared across multiple accounts (because the hashes are identical)

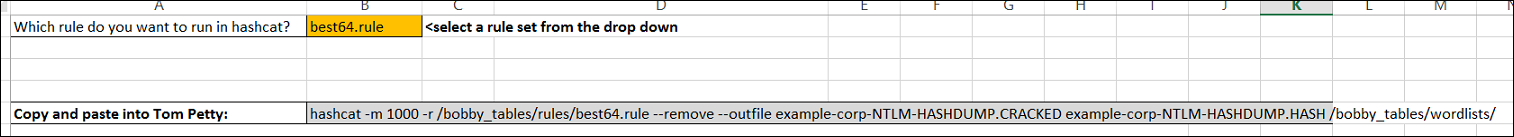
* 1. Click the next tab (“4. Hashcat format”) and you’ll see that the password hashes have been extracted and put into hashcat format:



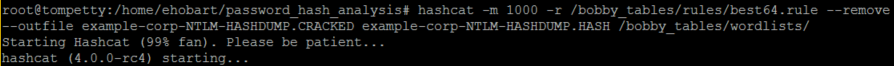
Create a blank file on the cracking server with the automatically computed filename, <clientname>-NTLM-HASHDUMP.HASH, and paste all of the hashcat formatted hashes into it.



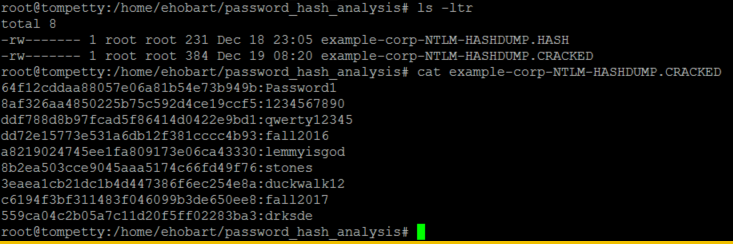
* 1. Click the next tab (“5. Hashcat command builder”). This tab has built out the entire hashcat command you’ll use on hashcat. It includes the automatically computed filenames from the previous tab. There’s also a drop down menu so you can choose which rule you want to use on hashcat.



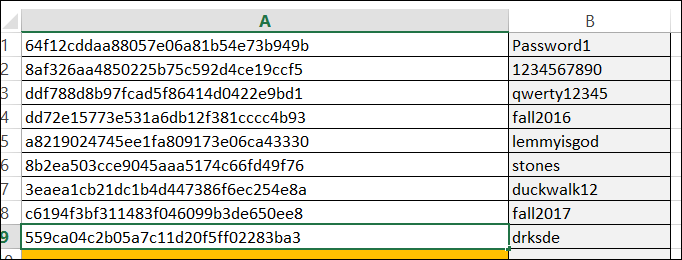
Copy and paste the command into the cracking server to begin cracking:



* 1. After you’ve cracked some passwords, you’ll notice that hashcat has output them neatly in hash:password format in a file named <clientname>-NTLM-HASHDUMP.CRACK:

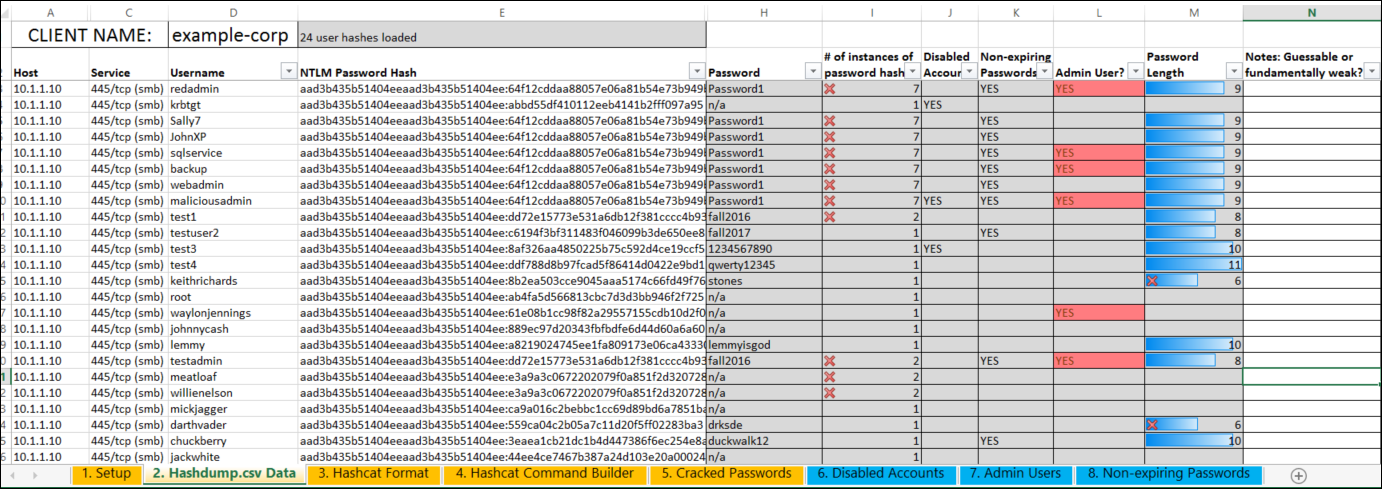


Goto the next tab (“5. Cracked passwords”) and paste this output in hash:password format into column A. The spreadsheet \***should**\* automatically split the data on the “:” character and place the cleartext password into column B. However, the Text to Column feature in Excel seems to act a bit wonky at times and doesn’t always do this automatically. If this occurs, select column A and click Text to Columns and split on the “:” character. Whether it’s been done it automatically, or you’ve used the workaround, the tab should look like this:



The rest of the tabs are optional but highly recommended.

* 1. Click the 6th tab (“6. Disabled Accounts”). Run the included PowerShell command on the compromised Domain Controller to extract a list of all of the disabled accounts in the domain. Paste the list into column A. Repeat this on tabs 7 and 8 with their respective included PowerShell commands to extract a list the domain admins and users with non-expiring passwords. Now the hashdump analysis tool has been filled in with data and can be used to identify trends.



Attached is a completed example: example-corp\_hashdump\_analysis\_v9.xlsx

Also attached is a blank version of the tool to be used on pentests: CLIENTNAME\_hashdump\_analysis\_v9.xlsx