Automating HalfLMChall Hash Cracking

written by Karl Fosaaen | November 9, 2012

Frequently during penetration tests, we will capture halflmchall password hashes from the network. These can come from a variety of sources, but common sources include NBNS spoofing and SQL queries/SQL injection. Both methods can be easy ways to get halflmchall hashes during a pen test.

For those who are unfamiliar with halflmchall hashes and how they are created, the process is pretty simple. When a client makes an authentication request with a server, the server responds with a challenge (which is basically a seed value used for hashing) for the client to use. If we are acting as the server, we can specify the challenge and the authentication protocol that we want to use. If we have a static challenge (1122334455667788) and a weak authentication protocol (LANMAN v1), we're able to quickly look up the captured hashes in rainbow tables. Now, this isn't the full technical detail of the process, but hopefully this gives you a good idea as to how this can work to our advantage.

For a much more in-depth review of the process, here's a great write up – http://www.foofus.net/?page_id=63 The typical capture to cracked password process goes like this:

1. Obtain a man-in-the-middle position, or force a server to authenticate to your evil server.
2. Capture the hash (via SMB or HTTP servers).
3. Look up the first 16 characters of the captured LM hash in the HalfLMChall rainbow tables.
4. Use the cracked portion of the LM hash to feed into the John the Ripper netntlm.pl script to crack the rest of the LM hash.
5. Feed the case insensitive password (from step 3) back

into the netntlm.pl script to crack the case sensitive NTLM hash and get the full password.

The cracking process goes pretty quickly, but it does require multiple commands to run that includes some copy and paste work. I've found that this process takes up more of my time than I would like, so I wrote a PowerShell script to automate the whole cracking process.

PowerShell cracking script requirements:

1. The halflmchall rainbow tables
2. Rcracki_mt
3. John the Ripper (Jumbo release)
4. Perl (required to run netntlm.pl)
5. You will also need to enable PowerShell to run scripts: "Set-ExecutionPolicy RemoteSigned"

Within the script you will have to specify your John, rcrack, Perl, and rainbow table locations, but you should be able to run the script from any directory.

The script usage is simple:

```
PS_MultiCrack.ps1 Input_File Output_File
```

You should be able to use the john formatted output file generated from the metasploit modules, but below is the basic format that the script will require:

```
DomainUser:::LMHASH:NTLMHASH:1122334455667788
ExampleTestAdmin:::daf4ce8f1965961138e76ee328e595e0c0c2d9a83fbe83fb:211af68207f7c88a1ad6c103a56966d1da1c1e91f02291f0:1122334455667788
```

The "1122334455667788" is the default static salt that is used by most of the tools used for capturing the hashes. It's also the salt used by the rainbow tables (Download here).

The script will write out each username and password to the output file when it's done. Hashes that are already in the john.pot file will be prefixed in your output file as “Previously Cracked:” so that you don’t have to worry about cleaning out your input file as you add more hashes. Additionally, the script won’t go through the effort of cracking the same hash again, as that would be a waste of time.

If you have any comments, suggestions, issues, please let me know through here or GitHub and I’ll try to address them.

GitHub Repo