Offline Password Auditing

Why Cover This?

- Demonstrate cracking of local passwords and hashes
  - Bad guys use this to escalate privileges on target hosts and other devices on the network
  - Can also be an authentication attack
- Best practices to keep users from exploiting these weaknesses
- Know your enemy

Why Crack Local Passwords?

- Escalate privileges on the local host
- Use local passwords to gain access to other systems
  - Admins often reuse passwords across network hosts
- Just for fun

Possible Scenario

- Bob uses a bootdisk to copy the SAM and SYSTEM files off his workstation
  - Or something like SAMDump if he's an admin
- Cracks them at home with John the Ripper
- Uses the local admin password he cracked to access admin privs on other machines
- Attempts to access other remote servers with those or other creds.

Password Cracking…

- Cracking a password is the act of finding the plaintext password by reversing the encryption method it's stored with
- Resetting a password != Cracking
  - Reset = quick smash and grab
  - Cracking = more stealthy, but takes time
- Unauthorized password cracking is illegal!

Windows Smash and Grab

- Nordahl – bootdisk we can use to reset local Windows passwords
  - http://pogostick.net/~pnh/ntpasswd/
- More or less tested from NT3.5 up to Windows 8.1, including the server versions like 2003, 2008 and 2012
  - Also 64 bit windows supported!
- Will not work if the disk is encrypted
  - If you reset a user's password that had encryption enabled, their files are gone…

chntpw

- Utility installed on Kali
- Live-boot Kali, we'll bypass some protections on the SAM file
- fdisk –l to list available partitions
- mount /dev/sda2 /mnt/sda2
- chntpw –I /mnt/sda2/Windows/system32/config/SAM
  - Starts chntpw in interactive mode

SAM Cracking Prevention

- These make security better, but maintenance more difficult…
  - Disable booting to anything but the hard drive in the BIOS
    - But someone could change that in the BOIS too
  - Put a password on the BIOS
    - Could pull the CMOS battery
    - Could pull the Hard Drive to a different computer
  - Lock the case
    - Can physical security on a case really be that good?
  - Epoxy the USB ports?
    - Well this got a bit extreme…

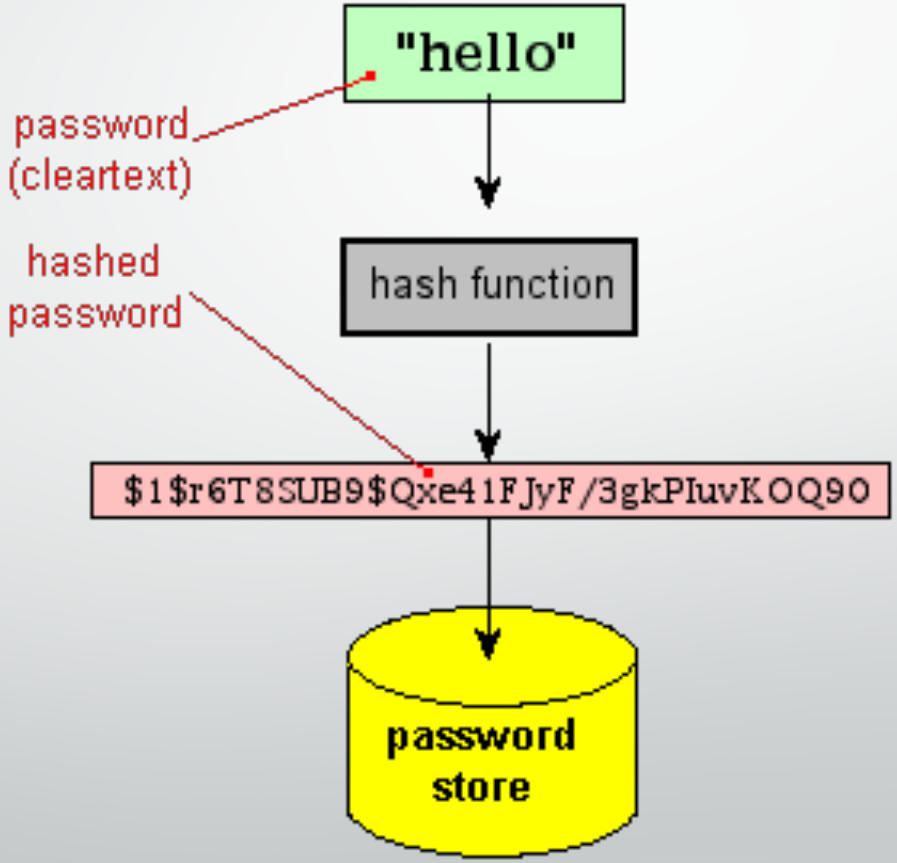
How are passwords stored?

- Windows and *most* websites don't store your password in plain text or a human readable word
  - If they do, and the DB is compromised, there are problems…
- Hashes
- Salted Passwords

Hashes

- Hashes are one-way, non-reversible versions of some data.
  - MD5, SHA256 sound familiar?
- To authenticate a user, their password is hashed, then compared with the hash stored in the database
  - The idea is you can't get a password back from just the hash

Hashing


"hello"
password
(cleartext)
hashed
password
hash function
1r6T8SUB9$Qxe41FJyF/3gkPIuvKOQ90
password
store

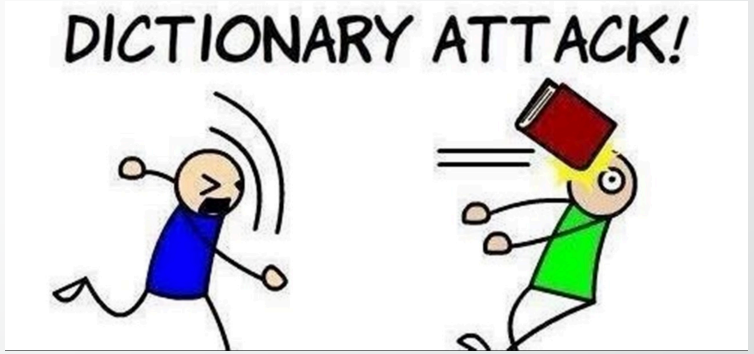
Some Sample Hashes

- A bunch of hashes here: http://openwall.info/wiki/john/sample-hashes
- LM
  - 855c3697d9979e78ac404c4ba2c66533
- NTLM
  - NT7f8fe03093cc84b267b109625f6bbf4b

Cracking Passwords – How Does This Work?

- Hashes are not easy to reverse – they were intentionally created with a one-way algorithm
- We must also create a hash to compare against.
  - We compare, when it matches, we found the password
- Our hash = "FSDglGo(*hSDF21wkDSF8vsw7"
  - "Baseball" → "7Ce1qlkjgt1gdbADFg124d"
  - "Password" → "1lkG9oiDo9igfds98oASFD"
  - "dog" → "FSDglGo(*hSDF21wkDSF8vsw7"
    - MATCH FOUND!

Two Methods…



- Dictionary Attack
  - Uses a list of words (dictionary) to try
  - Fast, but the password MUST be on the list for success
- Brute Force Attack
  - No preconfigured list
    - aaaa
    - aaab
    - aaac
    - …and so on
  - Slower, but, given enough time, you'll get the correct password

Linux Passwords

- Two files you need to know about…
  - /etc/passwd
    - the file where the user information (like username, user ID, group ID, location of home directory, login shell, …) is stored when a new user is created.
  - /etc/shadow
    - the file where important information (like an encrypted form of the password of a user, the day the password expires, whether or not the passwd has to be changed, the minimum and maximum time between password changes, …) is stored when a new user is created.

Shadow File

1. User name : It is your login name
2. Password: It your encrypted password. The password should be minimum 6-8 characters long including special characters/digits
3. Last password change (lastchanged): Days since Jan 1, 1970 that password was last changed
4. Minimum: The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change his/her password
5. Maximum: The maximum number of days the password is valid (after that user is forced to change his/her password)
6. Warn : The number of days before password is to expire that user is warned that his/her password must be changed
7. Inactive : The number of days after password expires that account is disabled
8. Expire : days since Jan 1, 1970 that account is disabled i.e. an absolute date specifying when the login may no longer be used

Passwd File

1. Username: It is used when user logs in. It should be between 1 and 32 characters in length
2. Password: An x character indicates that encrypted password is stored in /etc/shadow file
3. User ID (UID): Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups
4. Group ID (GID): The primary group ID (stored in /etc/group file)
5. User ID Info: The comment field. It allow you to add extra information about the users such as user's full name, phone number etc.
6. Home directory: The absolute path to the directory the user will be in when they log in
7. Command/shell: The absolute path of a command or shell (/bin/bash)

John the Ripper

- Free, highly effective, highly efficient, high speed, open source password cracking tool
  - Capable of cracking many different hash types
  - Highly customizable
- Flexible attack types:
  - Dictionary, brute force, hybrid
- Available for free:
  - http://www.openwall.com/john/

Using JtR

- john --user:<username> <hashFile>
- Note: The user name is case sensitive!

Using JtR

- john --user:<username> --wordlist:<dictionaryFile> <hashFile>

- /usr/share/wordlists/
- /usr/share/john/password.lst

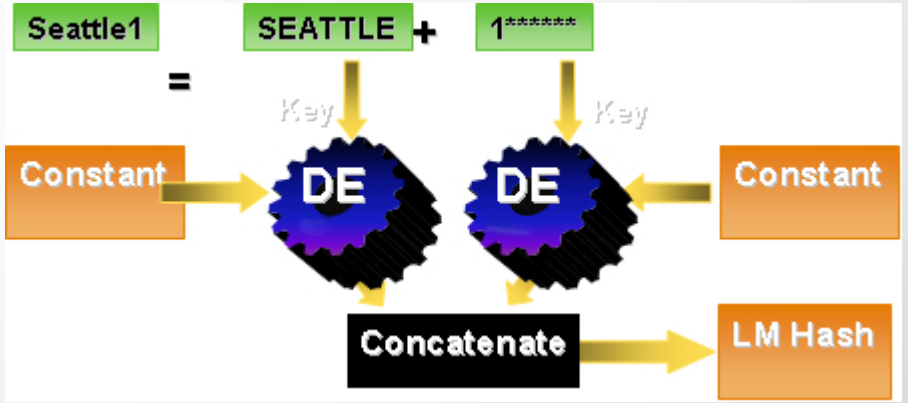
Unshadow

- Combines the passwd and shadow files so John can use them together.
- unshadow password-file shadow-file > linux_hashes.txt

How About Windows?

- Passwords stored in the SAM file (Security Accounts Manager) on NT based systems
- SAM normally found in…
  - %SYSTEMROOT%\System32\config\SAM
    - %SYSTEMROOT% == Windows Directory
      - Typically C:\Windows

LM Hashes



- LAN Manager
  - Used in older versions of Windows
    - Step 1: Convert to uppercase
      - Password1 = PASSWORD1
    - Step 2: Pad the plaintext with null chars to make it 14 bytes long
      - PASSWORD1 = PASSWORD1\o\o\o\o\o
    - Step 3: Split the password into two 7 byte/char chunks
      - PASSWOR D1\o\o\o\o\o
    - Step 4: Hash each chunk and concatenate
    - Step 5: Store in the SAM file

NTLM

- In an attempt to address the weaknesses of LM hashes, NTLM hashes were created
- Take unicode, mixed case password
- Utilize MD_4 to hash the password
- Store that hash in the SAM

NTLM Hashes

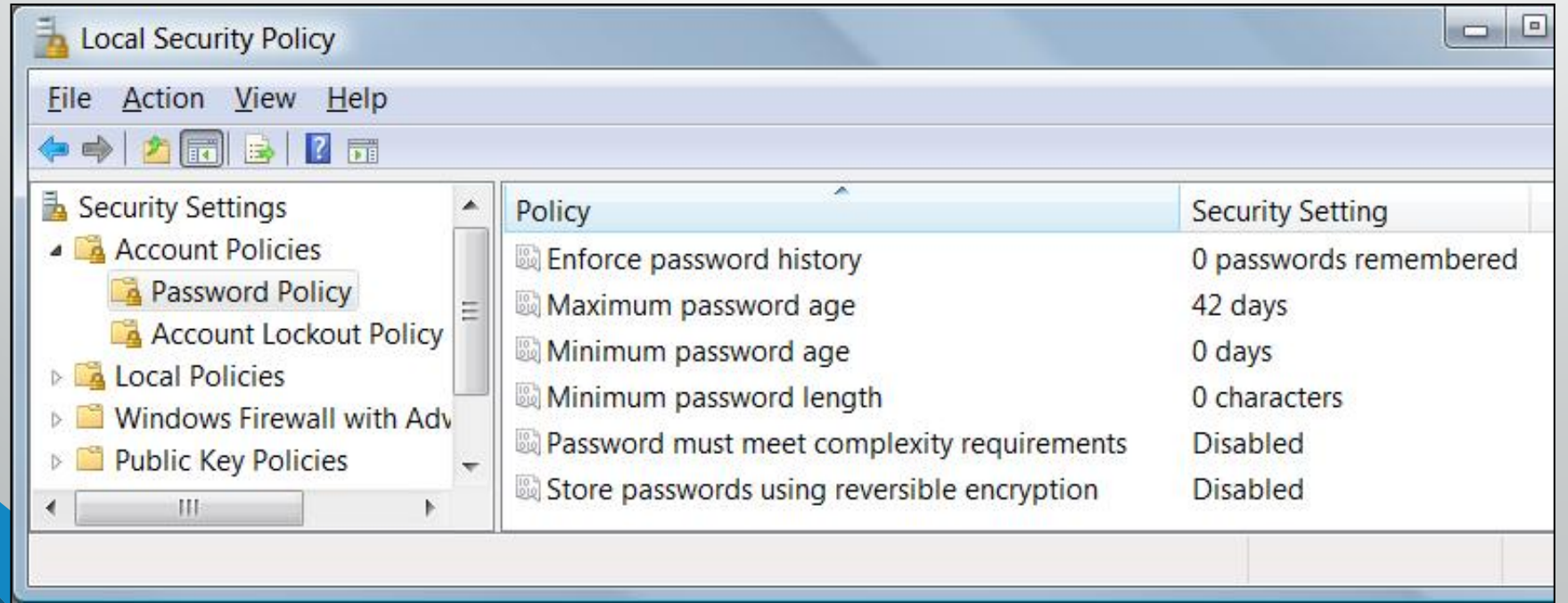
- Case sensitive
- Character set is 65,535
- Does not limit stored passwords to two 7-character parts
- Much stronger hashing algorithm

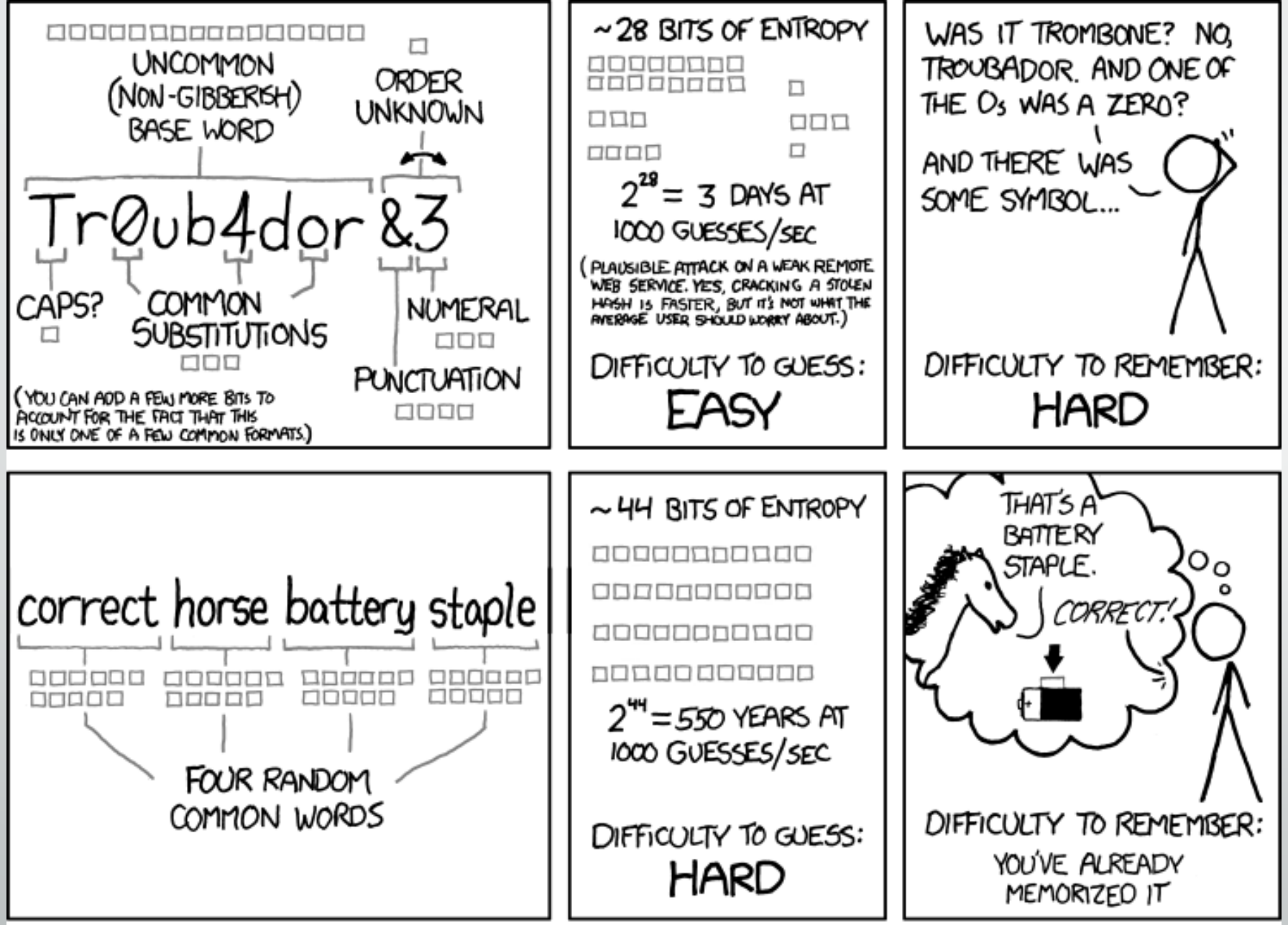
SAM Cracking Prevention

- Practical Methods / Fascist Methods
- Choose strong local passwords.
  - Use more then just alpha-numeric characters
    - If system allows: throw in some extended ASCII characters via Alt+numpad method
  - Turn off LM Hash Storage in the SAM via local policy, registry or via GPO
  - Use Passwords longer then 14 characters (passphrase)
  - Change the local password frequently
  - Use separate local admin passwords on public and staff boxes

Enforce a Strong Password Policy

- Control Panel → Administrative Tools → Computer Management → Local Security Policy → Account Policy → Password Policy
- Secpol.msc




UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor&3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS:
EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER:
HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS:
HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER:
YOU'VE ALREADY MEMORIZED IT
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

Dumping Hashes

```
hashdump
Administrator:500:e52cac67419a9a22ce171273f527391f:7facdc498ed1680c4fd1448319a8c04f:::
DSU:1003:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HelpAssistant:1000:f4063279bced32336908de0ff28c098f:9d9411eb7ed7e7db7d22d8bf923a65a6:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:45abdadb4029abd14768d5dcfdc0b478:::
meterpreter >
```

Questions?

- Assignment coming soon, I’m sure… Please Hold…