pentestmonkey
Taking the monkey work out of pentesting

- Site News
- Blog
- Tools
- Yaptest
- Cheat Sheets
- Contact

John The Ripper Hash Formats

John the Ripper is a favourite password cracking tool of many pentesters. There is plenty of documentation about its command line options.

I've encountered the following problems using John the Ripper. These are not problems with the tool itself, but inherent problems with pentesting and password cracking in general.

- Sometimes I stumble across hashes on a pentest, but don't recognise the format, don't know if it's supported by john, or whether there are multiple "–format" options I should try.
- The hashes you collect on a pentest sometimes need munging into a different format… but what's the format john is expecting?
- John will occasionally recognise your hashes as the wrong type (e.g. "Raw MD5″ as "LM DES"). This is inevitable because some hashes look identical.
- Sometimes I gain access to a system, but can't recall how to recover the password hashes for that particular application / OS.

These problems can all be sorted with a bit of googling or grepping through the john source code. I thought it might be helpful to compile a cheat sheet to reduce the amount of time I spend grepping and googling.

In the first release of this page I've:

- Copied example hashes out of the source code for most supported hash types.
- Provided examples of what your hashes.txt file might look like (though I'm sure other variations are supported that aren't covered here yet).
- For each example hash I've stated whether it will be automatically recognised by john, or whether you'll have to use the "–format" option (in which case I've included which –format option you need)

I haven't yet done the following:

- Added reminders on how hashes can be collected.
- Added information on how to munge the hashes into a format supported by john.

This sheet was originally based on john-1.7.8-jumbo-5. Changes in supported hashes or hash formats since then may not be reflected on this page.

afs – Kerberos AFS DES

Supported Hash Formats

```
$ cat hashes.txt
$K4$a8dc8aeaa2c48a97,
$ john hashes.txt
$ john --format=afs hashes.txt

$ cat hashes.txt
username:$K4$a8dc8aeaa2c48a97,
$ john hashes.txt
$ john --format=afs hashes.txt

$ cat hashes.txt
username:$K4$a8dc8aeaa2c48a97,:::::::
$ john hashes.txt
$ john --format=afs hashes.txt
```

bfegg – Eggdrop

Supported Hash Formats

```
$ cat hashes.txt
+C/.8o.Wuph9.
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=bfegg hashes.txt

$ cat hashes.txt
username:+C/.8o.Wuph9.
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=bfegg hashes.txt

$ cat hashes.txt
username:+C/.8o.Wuph9.:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=bfegg hashes.txt
```

bf – OpenBSD Blowfish

Supported Hash Formats

```
$ cat hashes.txt
$2a$05$CCCCCCCCCCCCCCCCCCCCCC.7uG0VCzI2bS7j6ymqJi9CdcdxiRTWNy
$ john hashes.txt
$ john --format=bf hashes.txt

$ cat hashes.txt
username:$2a$05$CCCCCCCCCCCCCCCCCCCCCC.7uG0VCzI2bS7j6ymqJi9CdcdxiRTWNy
$ john hashes.txt
$ john --format=bf hashes.txt

$ cat hashes.txt
username:$2a$05$CCCCCCCCCCCCCCCCCCCCCC.7uG0VCzI2bS7j6ymqJi9CdcdxiRTWNy:::::::
$ john hashes.txt
$ john --format=bf hashes.txt
```

bsdi – BSDI DES

Supported Hash Formats

```
$ cat hashes.txt
_J9..SDSD5YGyRCr4W4c
$ john hashes.txt
$ john --format=bsdi hashes.txt

$ cat hashes.txt
username:_J9..SDSD5YGyRCr4W4c
$ john hashes.txt
$ john --format=bsdi hashes.txt

$ cat hashes.txt
username:_J9..SDSD5YGyRCr4W4c:::::::
$ john hashes.txt
$ john --format=bsdi hashes.txt
```

crypt – generic crypt(3)

Supported Hash Formats

```
$ cat hashes.txt
SDbsugeBiC58A
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=crypt hashes.txt

$ cat hashes.txt
username:SDbsugeBiC58A
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=crypt hashes.txt
```

```
$ cat hashes.txt
username:SDbsugeBiC58A:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "Traditional DES".
$ john --format=crypt hashes.txt
```

des – Traditional DES

Supported Hash Formats

```
$ cat hashes.txt
SDbsugeBiC58A
$ john hashes.txt
$ john --format=des hashes.txt

$ cat hashes.txt
username:SDbsugeBiC58A
$ john hashes.txt
$ john --format=des hashes.txt

$ cat hashes.txt
username:SDbsugeBiC58A:::::::
$ john hashes.txt
$ john --format=des hashes.txt
```

dmd5 – DIGEST-MD5

Supported Hash Formats

TODO: No working example yet.

dominosec – More Secure Internet Password

Supported Hash Formats

```
$ cat hashes.txt
(GVMroLzc50YK/Yd+L8KH)
$ john hashes.txt
$ john --format=dominosec hashes.txt

$ cat hashes.txt
username:(GVMroLzc50YK/Yd+L8KH)
$ john hashes.txt
$ john --format=dominosec hashes.txt

$ cat hashes.txt
username:(GVMroLzc50YK/Yd+L8KH):::::::
$ john hashes.txt
$ john --format=dominosec hashes.txt
```

<none> – EPiServer SID Hashes

Supported Hash Formats

```
$ cat hashes.txt
0x5F1D84A6DE97E2BEFB637A3CB5318AFEF0750B856CF1836BD1D4470175BE 0x4D5EFDFA143EDF74193076F174AC47CEBF2F417F
$ john hashes.txt
$ # NB: There is no --format option for this hash type

$ cat hashes.txt
username:0x5F1D84A6DE97E2BEFB637A3CB5318AFEF0750B856CF1836BD1D4470175BE 0x4D5EFDFA143EDF74193076F174AC47CEBF2F417F
$ john hashes.txt
$ # NB: There is no --format option for this hash type
```

```
$ cat hashes.txt
username:0x5F1D84A6DE97E2BEFB637A3CB5318AFEF0750B856CF1836BD1D4470175BE 0x4D5EFDFA143EDF74193076F174AC47CEBF2F417F:::::::
$ john hashes.txt
$ # NB: There is no --format option for this hash type
```

hdaa – HTTP Digest access authentication

Supported Hash Formats

```
$ cat hashes.txt
$response$679066476e67b5c7c4e88f04be567f8b$user$myrealm$GET$/$8c12bd8f728afe56d45a0ce846b70e5a$00000001$4b61913cec32e2c9$auth
$ john hashes.txt
$ john --format=hdaa hashes.txt

$ cat hashes.txt
username:$response$679066476e67b5c7c4e88f04be567f8b$user$myrealm$GET$/$8c12bd8f728afe56d45a0ce846b70e5a$00000001$4b61913cec32e2c9$auth
$ john hashes.txt
$ john --format=hdaa hashes.txt

$ cat hashes.txt
username:$response$679066476e67b5c7c4e88f04be567f8b$user$myrealm$GET$/$8c12bd8f728afe56d45a0ce846b70e5a$00000001$4b61913cec32e2c9$auth:::::::
$ john hashes.txt
$ john --format=hdaa hashes.txt
```

hmac-md5 – HMAC MD5

Supported Hash Formats

```
$ cat hashes.txt
what do ya want for nothing?#750c783e6ab0b503eaa86e310a5db738
$ john hashes.txt
$ john --format=hmac-md5 hashes.txt

$ cat hashes.txt
username:what do ya want for nothing?#750c783e6ab0b503eaa86e310a5db738
$ john hashes.txt
$ john --format=hmac-md5 hashes.txt

$ cat hashes.txt
username:what do ya want for nothing?#750c783e6ab0b503eaa86e310a5db738:::::::
$ john hashes.txt
$ john --format=hmac-md5 hashes.txt
```

hmailserver – hmailserver

Supported Hash Formats

```
$ cat hashes.txt
cc06fa688a64cdeea43d3c0fb761fede7e3ccf00a9daea9c79f7d458e06f88327f16dd
$ john hashes.txt
$ john --format=hmailserver hashes.txt

$ cat hashes.txt
username:cc06fa688a64cdeea43d3c0fb761fede7e3ccf00a9daea9c79f7d458e06f88327f16dd
$ john hashes.txt
$ john --format=hmailserver hashes.txt

$ cat hashes.txt
username:cc06fa688a64cdeea43d3c0fb761fede7e3ccf00a9daea9c79f7d458e06f88327f16dd:::::::
$ john hashes.txt
$ john --format=hmailserver hashes.txt
```

ipb2 – IPB2 MD5

Supported Hash Formats

```
$ cat hashes.txt
$IPB2$2e75504633$d891f03a7327639bc632d62a7f302604
$ john hashes.txt
$ john --format=ipb2 hashes.txt

$ cat hashes.txt
username:$IPB2$2e75504633$d891f03a7327639bc632d62a7f302604
$ john hashes.txt
$ john --format=ipb2 hashes.txt

$ cat hashes.txt
username:$IPB2$2e75504633$d891f03a7327639bc632d62a7f302604:::::::
$ john hashes.txt
$ john --format=ipb2 hashes.txt
```

krb4 – Kerberos v4 TGT

Supported Hash Formats

```
$ cat hashes.txt
$af$ENGIN.UMICH.EDU$44feffd06e68e30bc8890e253760858d
$ john hashes.txt
$ john --format=krb4 hashes.txt

$ cat hashes.txt
username:$af$ENGIN.UMICH.EDU$44feffd06e68e30bc8890e253760858d
$ john hashes.txt
$ john --format=krb4 hashes.txt

$ cat hashes.txt
username:$af$ENGIN.UMICH.EDU$44feffd06e68e30bc8890e253760858d:::::::
$ john hashes.txt
$ john --format=krb4 hashes.txt
```

krb5 – Kerberos v5 TGT

Supported Hash Formats

```
$ cat hashes.txt
$krb5$oskov$ACM.UIUC.EDU$4730d7249765615d6f3652321c4fb76d09fb9cd06faeb0c31b8737f9fdfcde4bd4259c31cb1dff25df39173b09abdff08373302d99ac09802a290915243d9f0ea0313fdedc7f8d1fae0d9df8f0ee6233818d317f03a72c2
$ john hashes.txt
$ john --format=krb5 hashes.txt

$ cat hashes.txt
username:$krb5$oskov$ACM.UIUC.EDU$4730d7249765615d6f3652321c4fb76d09fb9cd06faeb0c31b8737f9fdfcde4bd4259c31cb1dff25df39173b09abdff08373302d99ac09802a290915243d9f0ea0313fdedc7f8d1fae0d9df8f0ee6233818d31
$ john hashes.txt
$ john --format=krb5 hashes.txt

$ cat hashes.txt
username:$krb5$oskov$ACM.UIUC.EDU$4730d7249765615d6f3652321c4fb76d09fb9cd06faeb0c31b8737f9fdfcde4bd4259c31cb1dff25df39173b09abdff08373302d99ac09802a290915243d9f0ea0313fdedc7f8d1fae0d9df8f0ee6233818d31
$ john hashes.txt
$ john --format=krb5 hashes.txt
```

lm – LM DES

Supported Hash Formats

```
$ cat hashes.txt
$LM$a9c604d244c4e99d
$ john hashes.txt
$ john --format=lm hashes.txt
```

```
$ cat hashes.txt
username:$LM$a9c604d244c4e99d
$ john hashes.txt
$ john --format=lm hashes.txt

$ cat hashes.txt
username:$LM$a9c604d244c4e99d:::::::
$ john hashes.txt
$ john --format=lm hashes.txt
```

lotus5 – Lotus5

Supported Hash Formats

```
$ cat hashes.txt
355E98E7C7B59BD810ED845AD0FD2FC4
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=lotus5 hashes.txt

$ cat hashes.txt
username:355E98E7C7B59BD810ED845AD0FD2FC4
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=lotus5 hashes.txt

$ cat hashes.txt
username:355E98E7C7B59BD810ED845AD0FD2FC4:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=lotus5 hashes.txt
```

md4-gen – Generic salted MD4

Supported Hash Formats

```
$ cat hashes.txt
$MD4p$salt$15ad2b7a23e5088942f9d3772181b384
$ john hashes.txt
$ john --format=md4-gen hashes.txt

$ cat hashes.txt
username:$MD4p$salt$15ad2b7a23e5088942f9d3772181b384
$ john hashes.txt
$ john --format=md4-gen hashes.txt

$ cat hashes.txt
username:$MD4p$salt$15ad2b7a23e5088942f9d3772181b384:::::::
$ john hashes.txt
$ john --format=md4-gen hashes.txt
```

md5 – FreeBSD MD5

Supported Hash Formats

```
$ cat hashes.txt
$1$12345678$aIccj83HRDBo6ux1bVx7D1
$ john hashes.txt
$ john --format=md5 hashes.txt

$ cat hashes.txt
username:$1$12345678$aIccj83HRDBo6ux1bVx7D1
$ john hashes.txt
$ john --format=md5 hashes.txt

$ cat hashes.txt
username:$1$12345678$aIccj83HRDBo6ux1bVx7D1:::::::
$ john hashes.txt
$ john --format=md5 hashes.txt
```

```
$ cat hashes.txt
$apr1$Q6ZYh...$RV6ft2bZ8j.NGrxLYaJt9.
$ john hashes.txt
$ john --format=md5 hashes.txt

$ cat hashes.txt
username:$apr1$Q6ZYh...$RV6ft2bZ8j.NGrxLYaJt9.
$ john hashes.txt
$ john --format=md5 hashes.txt

$ cat hashes.txt
username:$apr1$Q6ZYh...$RV6ft2bZ8j.NGrxLYaJt9.:::::::
$ john hashes.txt
$ john --format=md5 hashes.txt
```

md5-gen – Generic MD5

Supported Hash Formats

TODO: No working example yet.

mediawiki – MediaWiki MD5s

Supported Hash Formats

```
$ cat hashes.txt
$B$113$de2874e33da25313d808d2a8cbf31485
$ john hashes.txt
$ john --format=mediawiki hashes.txt

$ cat hashes.txt
username:$B$113$de2874e33da25313d808d2a8cbf31485
$ john hashes.txt
$ john --format=mediawiki hashes.txt

$ cat hashes.txt
username:$B$113$de2874e33da25313d808d2a8cbf31485:::::::
$ john hashes.txt
$ john --format=mediawiki hashes.txt
```

mscash – M$ Cache Hash

Supported Hash Formats

```
$ cat hashes.txt
M$test1#64cd29e36a8431a2b111378564a10631
$ john hashes.txt # Doesn't work.  JTR detects hash as "HMAC MD5".
$ john --format=mscash hashes.txt

$ cat hashes.txt
username:M$test1#64cd29e36a8431a2b111378564a10631
$ john hashes.txt # Doesn't work.  JTR detects hash as "HMAC MD5".
$ john --format=mscash hashes.txt

$ cat hashes.txt
username:M$test1#64cd29e36a8431a2b111378564a10631:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "HMAC MD5".
$ john --format=mscash hashes.txt
```

mscash2 – M$ Cache Hash 2 (DCC2)

Supported Hash Formats

```
$ cat hashes.txt
$DCC2$10240#test1#607bbe89611e37446e736f7856515bf8
$ john hashes.txt # Doesn't work.  JTR detects hash as "M$ Cache Hash".
$ john --format=mscash2 hashes.txt

$ cat hashes.txt
username:$DCC2$10240#test1#607bbe89611e37446e736f7856515bf8
$ john hashes.txt
$ john --format=mscash2 hashes.txt

$ cat hashes.txt
username:$DCC2$10240#test1#607bbe89611e37446e736f7856515bf8:::::::
$ john hashes.txt
$ john --format=mscash2 hashes.txt
```

mschapv2 – MSCHAPv2 C/R MD4 DES

Supported Hash Formats

```
$ cat hashes.txt
$MSCHAPv2$d94e7c7972b2376b28c268583e162de7$eba25a3b04d2c7085d01f842e2befc91745c40db0f792356$0677ca7318fd7f65ae1b4f58c9f4f400$lameuser
$ john hashes.txt
$ john --format=mschapv2 hashes.txt

$ cat hashes.txt
username:$MSCHAPv2$d94e7c7972b2376b28c268583e162de7$eba25a3b04d2c7085d01f842e2befc91745c40db0f792356$0677ca7318fd7f65ae1b4f58c9f4f400$lameuser
$ john hashes.txt
$ john --format=mschapv2 hashes.txt

$ cat hashes.txt
username:$MSCHAPv2$d94e7c7972b2376b28c268583e162de7$eba25a3b04d2c7085d01f842e2befc91745c40db0f792356$0677ca7318fd7f65ae1b4f58c9f4f400$lameuser:::::::
$ john hashes.txt
$ john --format=mschapv2 hashes.txt
```

mskrb5 – MS Kerberos 5 AS-REQ Pre-Auth

Supported Hash Formats

```
$ cat hashes.txt
$mskrb5$$$98cd00b6f222d1d34e08fe0823196e0b$5937503ec29e3ce4e94a051632d0fff7b6781f93e3decf7dca707340239300d602932154
$ john hashes.txt
$ john --format=mskrb5 hashes.txt

$ cat hashes.txt
username:$mskrb5$$$98cd00b6f222d1d34e08fe0823196e0b$5937503ec29e3ce4e94a051632d0fff7b6781f93e3decf7dca707340239300d602932154
$ john hashes.txt
$ john --format=mskrb5 hashes.txt

$ cat hashes.txt
username:$mskrb5$$$98cd00b6f222d1d34e08fe0823196e0b$5937503ec29e3ce4e94a051632d0fff7b6781f93e3decf7dca707340239300d602932154:::::::
$ john hashes.txt
$ john --format=mskrb5 hashes.txt
```

mssql05 – MS-SQL05

Supported Hash Formats

```
$ cat hashes.txt
0x01004086CEB6BF932BC4151A1AF1F13CD17301D70816A8886908
$ john hashes.txt
$ john --format=mssql05 hashes.txt

$ cat hashes.txt
username:0x01004086CEB6BF932BC4151A1AF1F13CD17301D70816A8886908
$ john hashes.txt
$ john --format=mssql05 hashes.txt
```

```
$ cat hashes.txt
username:0x01004086CEB6BF932BC4151A1AF1F13CD17301D70816A8886908:::::::
$ john hashes.txt
$ john --format=mssql05 hashes.txt
```

mssql – MS-SQL

Supported Hash Formats

```
$ cat hashes.txt
0x0100A607BA7C54A24D17B565C59F1743776A10250F581D482DA8B6D6261460D3F53B279CC6913CE747006A2E3254
$ john hashes.txt
$ john --format=mssql hashes.txt

$ cat hashes.txt
username:0x0100A607BA7C54A24D17B565C59F1743776A10250F581D482DA8B6D6261460D3F53B279CC6913CE747006A2E3254
$ john hashes.txt
$ john --format=mssql hashes.txt

$ cat hashes.txt
username:0x0100A607BA7C54A24D17B565C59F1743776A10250F581D482DA8B6D6261460D3F53B279CC6913CE747006A2E3254:::::::
$ john hashes.txt
$ john --format=mssql hashes.txt
```

mysql-fast – MYSQL_fast

Supported Hash Formats

```
$ cat hashes.txt
60671c896665c3fa
$ john hashes.txt
$ john --format=mysql-fast hashes.txt

$ cat hashes.txt
username:60671c896665c3fa
$ john hashes.txt
$ john --format=mysql-fast hashes.txt

$ cat hashes.txt
username:60671c896665c3fa:::::::
$ john hashes.txt
$ john --format=mysql-fast hashes.txt
```

mysql – MYSQL

Supported Hash Formats

```
$ cat hashes.txt
5d2e19393cc5ef67
$ john hashes.txt # Doesn't work.  JTR detects hash as "MYSQL_fast".
$ john --format=mysql hashes.txt

$ cat hashes.txt
username:5d2e19393cc5ef67
$ john hashes.txt # Doesn't work.  JTR detects hash as "MYSQL_fast".
$ john --format=mysql hashes.txt

$ cat hashes.txt
username:5d2e19393cc5ef67:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "MYSQL_fast".
$ john --format=mysql hashes.txt
```

mysql-sha1 – MySQL 4.1 double-SHA-1

Supported Hash Formats

```
$ cat hashes.txt
*2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19
$ john hashes.txt
$ john --format=mysql-sha1 hashes.txt

$ cat hashes.txt
username:*2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19
$ john hashes.txt
$ john --format=mysql-sha1 hashes.txt

$ cat hashes.txt
username:*2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19:::::::
$ john hashes.txt
$ john --format=mysql-sha1 hashes.txt
```

netlm – LM C/R DES

Supported Hash Formats

```
$ cat hashes.txt
$NETLM$1122334455667788$0836F085B124F33895875FB1951905DD2F85252CC731BB25
$ john hashes.txt
$ john --format=netlm hashes.txt

$ cat hashes.txt
username:$NETLM$1122334455667788$0836F085B124F33895875FB1951905DD2F85252CC731BB25
$ john hashes.txt
$ john --format=netlm hashes.txt

$ cat hashes.txt
username:$NETLM$1122334455667788$0836F085B124F33895875FB1951905DD2F85252CC731BB25:::::::
$ john hashes.txt
$ john --format=netlm hashes.txt
```

netlmv2 – LMv2 C/R MD4 HMAC-MD5

Supported Hash Formats

```
$ cat hashes.txt
$NETLMv2$USER1$1122334455667788$B1D163EA5881504F3963DC50FCDC26C1$EB4D9E8138149E20
$ john hashes.txt
$ john --format=netlmv2 hashes.txt

$ cat hashes.txt
username:$NETLMv2$USER1$1122334455667788$B1D163EA5881504F3963DC50FCDC26C1$EB4D9E8138149E20
$ john hashes.txt
$ john --format=netlmv2 hashes.txt

$ cat hashes.txt
username:$NETLMv2$USER1$1122334455667788$B1D163EA5881504F3963DC50FCDC26C1$EB4D9E8138149E20:::::::
$ john hashes.txt
$ john --format=netlmv2 hashes.txt
```

netntlm – NTLMv1 C/R MD4 DES [ESS MD5]

Supported Hash Formats

```
$ cat hashes.txt
$NETNTLM$1122334455667788$B2B2220790F40C88BCFF347C652F67A7C4A70D3BEBD70233
$ john hashes.txt
$ john --format=netntlm hashes.txt
```

```
$ cat hashes.txt
username:$NETNTLM$1122334455667788$B2B2220790F40C88BCFF347C652F67A7C4A70D3BEBD70233
$ john hashes.txt
$ john --format=netntlm hashes.txt

$ cat hashes.txt
username:$NETNTLM$1122334455667788$B2B2220790F40C88BCFF347C652F67A7C4A70D3BEBD70233:::::::
$ john hashes.txt
$ john --format=netntlm hashes.txt
```

netntlmv2 – NTLMv2 C/R MD4 HMAC-MD5

Supported Hash Formats

```
$ cat hashes.txt
$NETNTLMv2$NTLMV2TESTWORKGROUP$1122334455667788$07659A550D5E9D02996DFD95C87EC1D5$0101000000000000006CF6385B74CA01B3610B02D99732DD000000000200120057004F0052004B00470052004F005500500000010020004400410054005
$ john hashes.txt
$ john --format=netntlmv2 hashes.txt

$ cat hashes.txt
username:$NETNTLMv2$NTLMV2TESTWORKGROUP$1122334455667788$07659A550D5E9D02996DFD95C87EC1D5$0101000000000000006CF6385B74CA01B3610B02D99732DD000000000200120057004F0052004B00470052004F0055005005000010020004
$ john hashes.txt
$ john --format=netntlmv2 hashes.txt

$ cat hashes.txt
username:$NETNTLMv2$NTLMV2TESTWORKGROUP$1122334455667788$07659A550D5E9D02996DFD95C87EC1D5$0101000000000000006CF6385B74CA01B3610B02D99732DD000000000200120057004F0052004B00470052004F0055005005000010020004
$ john hashes.txt
$ john --format=netntlmv2 hashes.txt
```

nethalflm – HalfLM C/R DES

Supported Hash Formats

```
$ cat hashes.txt
$NETHALFLM$1122334455667788$6E1EC36D3417CE9E09A4424309F116C4C991948DAEB4ADAD
$ john hashes.txt
$ john --format=nethalflm hashes.txt

$ cat hashes.txt
username:$NETHALFLM$1122334455667788$6E1EC36D3417CE9E09A4424309F116C4C991948DAEB4ADAD
$ john hashes.txt
$ john --format=nethalflm hashes.txt

$ cat hashes.txt
username:$NETHALFLM$1122334455667788$6E1EC36D3417CE9E09A4424309F116C4C991948DAEB4ADAD:::::::
$ john hashes.txt
$ john --format=nethalflm hashes.txt
```

md5ns – Netscreen MD5

Supported Hash Formats

```
$ cat hashes.txt
admin$nMjFM0rdC9iOc+xIFsGEm3LtAeGZhn
$ john hashes.txt
$ john --format=md5ns hashes.txt

$ cat hashes.txt
username:admin$nMjFM0rdC9iOc+xIFsGEm3LtAeGZhn
$ john hashes.txt
$ john --format=md5ns hashes.txt

$ cat hashes.txt
username:admin$nMjFM0rdC9iOc+xIFsGEm3LtAeGZhn:::::::
$ john hashes.txt
$ john --format=md5ns hashes.txt
```

nsldap – Netscape LDAP SHA

Supported Hash Formats

```
$ cat hashes.txt
{SHA}cMiB1KJphN3OeV9vcYF8nPRIDnk=
$ john hashes.txt
$ john --format=nsldap hashes.txt

$ cat hashes.txt
username:{SHA}cMiB1KJphN3OeV9vcYF8nPRIDnk=
$ john hashes.txt
$ john --format=nsldap hashes.txt

$ cat hashes.txt
username:{SHA}cMiB1KJphN3OeV9vcYF8nPRIDnk=:::::::
$ john hashes.txt
$ john --format=nsldap hashes.txt
```

ssha – Netscape LDAP SSHA

Supported Hash Formats

```
$ cat hashes.txt
{SSHA}WTT3B9Jjr8gOt0Q7WMs9/XvukyhTQj0Ns0jMKQ==
$ john hashes.txt
$ john --format=ssha hashes.txt

$ cat hashes.txt
username:{SSHA}WTT3B9Jjr8gOt0Q7WMs9/XvukyhTQj0Ns0jMKQ==
$ john hashes.txt
$ john --format=ssha hashes.txt

$ cat hashes.txt
username:{SSHA}WTT3B9Jjr8gOt0Q7WMs9/XvukyhTQj0Ns0jMKQ==:::::::
$ john hashes.txt
$ john --format=ssha hashes.txt
```

nt – NT MD4

Supported Hash Formats

```
$ cat hashes.txt
$NT$8846f7eaee8fb117ad06bdd830b7586c
$ john hashes.txt
$ john --format=nt hashes.txt

$ cat hashes.txt
username:$NT$8846f7eaee8fb117ad06bdd830b7586c
$ john hashes.txt
$ john --format=nt hashes.txt

$ cat hashes.txt
username:$NT$8846f7eaee8fb117ad06bdd830b7586c:::::::
$ john hashes.txt
$ john --format=nt hashes.txt
```

openssha – OpenLDAP SSHA

Supported Hash Formats

```
$ cat hashes.txt
{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X
```

```
$ john hashes.txt
$ john --format=openssha hashes.txt

$ cat hashes.txt
username:{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X
$ john hashes.txt
$ john --format=openssha hashes.txt

$ cat hashes.txt
username:{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X:::::::
$ john hashes.txt
$ john --format=openssha hashes.txt
```

oracle11 – Oracle 11g

Supported Hash Formats

```
$ cat hashes.txt
5FDAB69F543563582BA57894FE1C1361FB8ED57B903603F2C52ED1B4D642
$ john hashes.txt
$ john --format=oracle11 hashes.txt

$ cat hashes.txt
username:5FDAB69F543563582BA57894FE1C1361FB8ED57B903603F2C52ED1B4D642
$ john hashes.txt
$ john --format=oracle11 hashes.txt

$ cat hashes.txt
username:5FDAB69F543563582BA57894FE1C1361FB8ED57B903603F2C52ED1B4D642:::::::
$ john hashes.txt
$ john --format=oracle11 hashes.txt
```

oracle – Oracle

Supported Hash Formats

```
$ cat hashes.txt
O$SIMON#4F8BC1809CB2AF77
$ john hashes.txt
$ john --format=oracle hashes.txt

$ cat hashes.txt
username:O$SIMON#4F8BC1809CB2AF77
$ john hashes.txt
$ john --format=oracle hashes.txt

$ cat hashes.txt
username:O$SIMON#4F8BC1809CB2AF77:::::::
$ john hashes.txt
$ john --format=oracle hashes.txt
```

pdf – pdf

Supported Hash Formats

```
$ cat hashes.txt
$pdf$Standard*badad1e86442699427116d3e5d5271bc80a27814fc5e80f815efeef839354c5f*289ece9b5ce451a5d7064693dab3badf101112131415161718191a1b1c1d1e1f*16*34b1b6e593787af681a9b63fa8bf563b*1*1*0*1*4*128*-4*3*2
$ john hashes.txt
$ john --format=pdf hashes.txt

$ cat hashes.txt
username:$pdf$Standard*badad1e86442699427116d3e5d5271bc80a27814fc5e80f815efeef839354c5f*289ece9b5ce451a5d7064693dab3badf101112131415161718191a1b1c1d1e1f*16*34b1b6e593787af681a9b63fa8bf563b*1*1*0*1*4*
$ john hashes.txt
$ john --format=pdf hashes.txt
```

```
$ cat hashes.txt
username:$pdf$Standard*badad1e86442699427116d3e5d5271bc80a27814fc5e80f815efeef839354c5f*289ece9b5ce451a5d7064693dab3badf101112131415161718191a1b1c1d1e1f*16*34b1b6e593787af681a9b63fa8bf563b*1*1*0*1*4*1
$ john hashes.txt
$ john --format=pdf hashes.txt
```

phpass-md5 – PHPass MD5

Supported Hash Formats

```
$ cat hashes.txt
$H$9aaaaaSXBjgypwqm.JsMssPLiS8YQ00
$ john hashes.txt
$ john --format=phpass-md5 hashes.txt

$ cat hashes.txt
username:$H$9aaaaaSXBjgypwqm.JsMssPLiS8YQ00
$ john hashes.txt
$ john --format=phpass-md5 hashes.txt

$ cat hashes.txt
username:$H$9aaaaaSXBjgypwqm.JsMssPLiS8YQ00:::::::
$ john hashes.txt
$ john --format=phpass-md5 hashes.txt
```

phps – PHPS MD5

Supported Hash Formats

```
$ cat hashes.txt
$PHPS$433925$5d756853cd63acee76e6dcd6d3728447
$ john hashes.txt
$ john --format=phps hashes.txt

$ cat hashes.txt
username:$PHPS$433925$5d756853cd63acee76e6dcd6d3728447
$ john hashes.txt
$ john --format=phps hashes.txt

$ cat hashes.txt
username:$PHPS$433925$5d756853cd63acee76e6dcd6d3728447:::::::
$ john hashes.txt
$ john --format=phps hashes.txt
```

pix-md5 – PIX MD5

Supported Hash Formats

```
$ cat hashes.txt
NuLKvvWGg.x9HEKO
$ john hashes.txt
$ john --format=pix-md5 hashes.txt

$ cat hashes.txt
username:NuLKvvWGg.x9HEKO
$ john hashes.txt
$ john --format=pix-md5 hashes.txt

$ cat hashes.txt
username:NuLKvvWGg.x9HEKO:::::::
$ john hashes.txt
$ john --format=pix-md5 hashes.txt
```

po – Post.Office MD5

Supported Hash Formats

```
$ cat hashes.txt
0c78bdef7d5448105cfbbc9aaa490a44550c41c11bab48f9dbd8203ed313eef0
$ john hashes.txt
$ john --format=po hashes.txt

$ cat hashes.txt
username:0c78bdef7d5448105cfbbc9aaa490a44550c41c11bab48f9dbd8203ed313eef0
$ john hashes.txt
$ john --format=po hashes.txt

$ cat hashes.txt
username:0c78bdef7d5448105cfbbc9aaa490a44550c41c11bab48f9dbd8203ed313eef0:::::::
$ john hashes.txt
$ john --format=po hashes.txt
```

rar – rar

Supported Hash Formats

```
$ cat hashes.txt
$rar3$*0*c9dea41b149b53b4*fcbdb66122d8ebdb32532c22ca7ab9ec*24
$ john hashes.txt
$ john --format=rar hashes.txt

$ cat hashes.txt
username:$rar3$*0*c9dea41b149b53b4*fcbdb66122d8ebdb32532c22ca7ab9ec*24
$ john hashes.txt
$ john --format=rar hashes.txt

$ cat hashes.txt
username:$rar3$*0*c9dea41b149b53b4*fcbdb66122d8ebdb32532c22ca7ab9ec*24:::::::
$ john hashes.txt
$ john --format=rar hashes.txt
```

raw-md4 – Raw MD4

Supported Hash Formats

```
$ cat hashes.txt
8a9d093f14f8701df17732b2bb182c74
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md4 hashes.txt

$ cat hashes.txt
username:8a9d093f14f8701df17732b2bb182c74
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md4 hashes.txt

$ cat hashes.txt
username:8a9d093f14f8701df17732b2bb182c74:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md4 hashes.txt
```

raw-md5 – Raw MD5

Supported Hash Formats

```
$ cat hashes.txt
5a105e8b9d40e1329780d62ea2265d8a
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5 hashes.txt
```

```
$ cat hashes.txt
username:5a105e8b9d40e1329780d62ea2265d8a
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5 hashes.txt

$ cat hashes.txt
username:5a105e8b9d40e1329780d62ea2265d8a:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5 hashes.txt
```

raw-md5-unicode – Raw MD5 of Unicode plaintext

Supported Hash Formats

```
$ cat hashes.txt
16c47151c18ac087cd12b3a70746c790
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5-unicode hashes.txt

$ cat hashes.txt
username:16c47151c18ac087cd12b3a70746c790
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5-unicode hashes.txt

$ cat hashes.txt
username:16c47151c18ac087cd12b3a70746c790:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "LM DES".
$ john --format=raw-md5-unicode hashes.txt
```

raw-sha1 – Raw SHA-1

Supported Hash Formats

```
$ cat hashes.txt
A9993E364706816ABA3E25717850C26C9CD0D89D
$ john hashes.txt
$ john --format=raw-sha1 hashes.txt

$ cat hashes.txt
username:A9993E364706816ABA3E25717850C26C9CD0D89D
$ john hashes.txt
$ john --format=raw-sha1 hashes.txt

$ cat hashes.txt
username:A9993E364706816ABA3E25717850C26C9CD0D89D:::::::
$ john hashes.txt
$ john --format=raw-sha1 hashes.txt
```

raw-sha224 – Raw SHA-224

Supported Hash Formats

```
$ cat hashes.txt
d63dc919e201d7bc4c825630d2cf25fdc93d4b2f0d46706d29038d01
$ john hashes.txt
$ john --format=raw-sha224 hashes.txt

$ cat hashes.txt
username:d63dc919e201d7bc4c825630d2cf25fdc93d4b2f0d46706d29038d01
$ john hashes.txt
$ john --format=raw-sha224 hashes.txt

$ cat hashes.txt
username:d63dc919e201d7bc4c825630d2cf25fdc93d4b2f0d46706d29038d01:::::::
$ john hashes.txt
$ john --format=raw-sha224 hashes.txt
```

raw-sha256 – Raw SHA-256

Supported Hash Formats

```
$ cat hashes.txt
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
$ john hashes.txt # Doesn't work.  JTR detects hash as "Post.Office MD5".
$ john --format=raw-sha256 hashes.txt

$ cat hashes.txt
username:5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
$ john hashes.txt # Doesn't work.  JTR detects hash as "Post.Office MD5".
$ john --format=raw-sha256 hashes.txt

$ cat hashes.txt
username:5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "Post.Office MD5".
$ john --format=raw-sha256 hashes.txt
```

raw-sha384 – Raw SHA-384

Supported Hash Formats

```
$ cat hashes.txt
a8b64babd0aca91a59bdbb7761b421d4f2bb38280d3a75ba0f21f2bebc45583d446c598660c94ce680c47d19c30783a7
$ john hashes.txt
$ john --format=raw-sha384 hashes.txt

$ cat hashes.txt
username:a8b64babd0aca91a59bdbb7761b421d4f2bb38280d3a75ba0f21f2bebc45583d446c598660c94ce680c47d19c30783a7
$ john hashes.txt
$ john --format=raw-sha384 hashes.txt

$ cat hashes.txt
username:a8b64babd0aca91a59bdbb7761b421d4f2bb38280d3a75ba0f21f2bebc45583d446c598660c94ce680c47d19c30783a7:::::::
$ john hashes.txt
$ john --format=raw-sha384 hashes.txt
```

raw-sha512 – Raw SHA-512

Supported Hash Formats

```
$ cat hashes.txt
b109f3bbbc244eb82441917ed06d618b9008dd09b3befd1b5e07394c706a8bb980b1d7785e5976ec049b46df5f1326af5a2ea6d103fd07c95385ffab0cacbc86
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt

$ cat hashes.txt
username:b109f3bbbc244eb82441917ed06d618b9008dd09b3befd1b5e07394c706a8bb980b1d7785e5976ec049b46df5f1326af5a2ea6d103fd07c95385ffab0cacbc86
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt

$ cat hashes.txt
username:b109f3bbbc244eb82441917ed06d618b9008dd09b3befd1b5e07394c706a8bb980b1d7785e5976ec049b46df5f1326af5a2ea6d103fd07c95385ffab0cacbc86:::::::
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt

$ cat hashes.txt
$SHA512$fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cddce372f139eff4d13959da6f1f5d3eabe
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt

$ cat hashes.txt
username:$SHA512$fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cddce372f139eff4d13959da6f1f5d3eabe
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt
```

```
$ cat hashes.txt
username:$SHA512$fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cddce372f139eff4d13959da6f1f5d3eabe:::::::
$ john hashes.txt
$ john --format=raw-sha512 hashes.txt
```

salted-sha – Salted SHA

Supported Hash Formats

```
$ cat hashes.txt
{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X
$ john hashes.txt # Doesn't work.  JTR detects hash as "OpenLDAP SSHA".
$ john --format=salted-sha hashes.txt

$ cat hashes.txt
username:{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X
$ john hashes.txt # Doesn't work.  JTR detects hash as "OpenLDAP SSHA".
$ john --format=salted-sha hashes.txt

$ cat hashes.txt
username:{SSHA}hHSEPW3qeiOo5Pl2MpHQCXh0vgfyVR/X:::::::
$ john hashes.txt # Doesn't work.  JTR detects hash as "OpenLDAP SSHA".
$ john --format=salted-sha hashes.txt
```

sapb – SAP BCODE

Supported Hash Formats

```
$ cat hashes.txt
ROOT                            $8366A4E9E6B72CB0
$ john hashes.txt
$ john --format=sapb hashes.txt

$ cat hashes.txt
username:ROOT                            $8366A4E9E6B72CB0
$ john hashes.txt
$ john --format=sapb hashes.txt

$ cat hashes.txt
username:ROOT                            $8366A4E9E6B72CB0:::::::
$ john hashes.txt
$ john --format=sapb hashes.txt
```

sapg – SAP CODVN G (PASSCODE)

Supported Hash Formats

```
$ cat hashes.txt
ROOT                            $1194E38F14B9F3F8DA1B181F14DEB70E7BDCC239
$ john hashes.txt
$ john --format=sapg hashes.txt

$ cat hashes.txt
username:ROOT                            $1194E38F14B9F3F8DA1B181F14DEB70E7BDCC239
$ john hashes.txt
$ john --format=sapg hashes.txt

$ cat hashes.txt
username:ROOT                            $1194E38F14B9F3F8DA1B181F14DEB70E7BDCC239:::::::
$ john hashes.txt
$ john --format=sapg hashes.txt
```

sha1-gen – Generic salted SHA-1

Supported Hash Formats

```
$ cat hashes.txt
$SHA1p$salt$59b3e8d637cf97edbe2384cf59cb7453dfe30789
$ john hashes.txt
$ john --format=sha1-gen hashes.txt

$ cat hashes.txt
username:$SHA1p$salt$59b3e8d637cf97edbe2384cf59cb7453dfe30789
$ john hashes.txt
$ john --format=sha1-gen hashes.txt

$ cat hashes.txt
username:$SHA1p$salt$59b3e8d637cf97edbe2384cf59cb7453dfe30789:::::::
$ john hashes.txt
$ john --format=sha1-gen hashes.txt
```

skey – S/Key

Supported Hash Formats

TODO: No working example yet.
TODO: No working example yet.
TODO: No working example yet.
TODO: No working example yet.

ssh – ssh

Supported Hash Formats

```
$ cat hashes.txt
$ssh2$2d2d2d2d2d424547494e204453412050524956415445204b45592d2d2d2d2d0a50726f632d547970653a20342c454e435259505445440a44454b2d496e666f3a204145532d3132382d4342432c35413830363832373943330463436453938323037
$ john hashes.txt
$ john --format=ssh hashes.txt

$ cat hashes.txt
username:$ssh2$2d2d2d2d2d424547494e204453412050524956415445204b45592d2d2d2d2d0a50726f632d547970653a20342c454e435259505445440a44454b2d496e666f3a204145532d3132382d4342432c35413830363832373943330463436453
$ john hashes.txt
$ john --format=ssh hashes.txt

$ cat hashes.txt
username:$ssh2$2d2d2d2d2d424547494e204453412050524956415445204b45592d2d2d2d2d0a50726f632d547970653a20342c454e435259505445440a44454b2d496e666f3a204145532d3132382d4342432c35413830363832373943330463436453
$ john hashes.txt
$ john --format=ssh hashes.txt
```

sybasease – sybasease

Supported Hash Formats

```
$ cat hashes.txt
0xc0074BE393C06BE420AD541671aa5e6f1a19a4a73bb51c59f45790f0887cfb70e0599747c6844d4556b3
$ john hashes.txt
$ john --format=sybasease hashes.txt

$ cat hashes.txt
username:0xc0074BE393C06BE420AD541671aa5e6f1a19a4a73bb51c59f45790f0887cfb70e0599747c6844d4556b3
$ john hashes.txt
$ john --format=sybasease hashes.txt

$ cat hashes.txt
username:0xc0074BE393C06BE420AD541671aa5e6f1a19a4a73bb51c59f45790f0887cfb70e0599747c6844d4556b3:::::::
$ john hashes.txt
$ john --format=sybasease hashes.txt
```

xsha – Mac OS X 10.4+ salted SHA-1

Supported Hash Formats

```
$ cat hashes.txt
12345678F9083C7F66F46A0A102E4CC17EC08C8AF120571B
$ john hashes.txt
$ john --format=xsha hashes.txt

$ cat hashes.txt
username:12345678F9083C7F66F46A0A102E4CC17EC08C8AF120571B
$ john hashes.txt
$ john --format=xsha hashes.txt

$ cat hashes.txt
username:12345678F9083C7F66F46A0A102E4CC17EC08C8AF120571B:::::::
$ john hashes.txt
$ john --format=xsha hashes.txt
```

zip – zip

Supported Hash Formats

```
$ cat hashes.txt
$zip$*0*1*8005b1b7d077708d*dee4
$ john hashes.txt
$ john --format=zip hashes.txt

$ cat hashes.txt
username:$zip$*0*1*8005b1b7d077708d*dee4
$ john hashes.txt
$ john --format=zip hashes.txt

$ cat hashes.txt
username:$zip$*0*1*8005b1b7d077708d*dee4:::::::
$ john hashes.txt
$ john --format=zip hashes.txt
```

Tags: johntheripper, pentest

Posted in Cheat Sheets

Categories

- Blog (78)
- Cheat Sheets (10)
  - Shells (1)
  - SQL Injection (7)
- Contact (2)
- Site News (3)
- Tools (17)
  - Audit (3)
  - Misc (7)
  - User Enumeration (4)
  - Web Shells (3)
- Uncategorized (3)
- Yaptest (15)
  - Front End (1)

- Installing (2)
- Overview (2)
- Using (8)

RSS Feed