

```

#!/usr/bin/perl
#
# K.Crean 10.02.2012, with thanks to Ben Rockwood's toasterview which got me started in
the right direction
# ckevinj@gmail.com

my $DEBUG=0;
if($ARGV[0] eq "" ){ printf("Usage: $0 hostname,hostname,hostname [options: -t=interval
(teamquest report)]\n"); exit; }

#### Configuration Variables that need to be set for your environment
my $SNMPGET = "/opt/csw/bin/snmpget";           # Location of snmpget command
my $SNMPTABLE = "/opt/csw/bin/snmpatable";     # Location of snmpatable command
my $MIB = "/opt/teamquest/scripts/data/netapp.mib";
        # Location of the NetApp MIB
my $COMMUNITY = "yourcommunity";             # Filer Community Name
my $SNMPuser="yourSNMPuser";                 # username on NetApp with the login-
snmp ability/role
my $SNMPup="yourSNMPuserPassword";
        # password for username on NetApp with the login-snm ability/role
my $snaps="no";                             # set to yes or no, to see snapshot data
my $aggronly="yes";                          # set to yes or no, to see only aggregate
data

my $SNMPOPTIONS="-v3 -n \"\" -u $SNMPuser -l authNoPriv -A $SNMPup -a Md5";
my $tqrep=0; my $interval=0;

#### Var setup
my @HOSTS= split (/,/, @ARGV[0]);

use Sys::Hostname;
    $host = hostname;
chomp(my $DATE = `date "+%m/%d/%Y %H:%M:00"`);

if ("${ARGV[1]}" =~ /^^-t/)
{
    $tqrep=1;
    ($nothing,$interval)= split /=/, $ARGV[1];
    if($DEBUG>0) { print STDERR "Teamquest report on at interval $interval.\n";}
    # Print TQ head
    print "\"$DATE\" $interval $host\n";
}
else { print "FileName VolumeType VolumeMount GBCapacity GBUsed GBFree PercentUsed\n"; }

foreach $HOST (@HOSTS)
{
    my $hostsumcap=0; my $hostsumused=0; my $hostsumpercent=0; my $hostsumfree=0;
    @dfoutput = `$SNMPTABLE $SNMPOPTIONS -c $COMMUNITY -C Hf \: -m $MIB $HOST NETWORK-
APPLIANCE-MIB::dfTable`;
    foreach(@dfoutput)
    {
        my $proceed=1;
        chomp($_);
        my @dfline = split(/:/);
        $dfline[1] =~ s/\\/g;           # Prune off the quotes on the vol name.
        if ($DEBUG>0){print STDERR "Working on dfline $dfline[1].\n";
            if ($DEBUG>1){foreach $debugged (@dfline){print STDERR "\t\t$debugged\n";}}
        }

        if (($dfline[1] =~ m/\.snapshot/)&&("$snaps" eq "no")){$proceed=0; if ($DEBUG>0){print
STDERR "\tCannot proceed.\n";}}
    }
}

```

```

if (($dfline[22] !~ m/aggregate/)&&("$aggronly" eq "yes")){$proceed=0; if ($DEBUG>0)
{print STDERR "\tCannot proceed.\n";}}
if (( $dfline[22] =~ m/aggregate/)&&($dfline[1] !~ m/\.snapshot/))
{
    $hostsumcap=$hostsumcap + (sprintf("%0.2f", $dfline[28] / 1048576 ));
    $hostsumused=$hostsumused + (sprintf("%0.2f", $dfline[29] / 1048576 ));
    $hostsumfree=$hostsumfree + (sprintf("%0.2f", $dfline[30] / 1048576 ));
    if ($DEBUG>0){print STDERR "Cap/Used: $hostsumcap/$hostsumused\n";}
}

## Check to proceed is the next test
## Cumulative Counters
if ($proceed>0)
{
    ### Round the numbers, from kbytes to gig
    $dfline[28] = sprintf("%0.2f", $dfline[28] / 1048576);
    $dfline[29] = sprintf("%0.2f", $dfline[29] / 1048576);
    $dfline[30] = sprintf("%0.2f", $dfline[30] / 1048576);

    # print TQ data
    print "\"$HOST\" \"$dfline[22]\" \"${dfline[1]}\" ${dfline[28]} ${dfline[29]} ${dfline
[30]} ${dfline[5]}\n";
}
}
$hostsumpercent= sprintf("%0.f", $hostsumused / $hostsumcap * 100 );
print "\"$HOST\" \"FilerCapacity\" \"-\" $hostsumcap $hostsumused $hostsumfree
$hostsumpercent\n";

}
print "\n";

```