Service setup

so, let's setup HTTP, FTP, SSH, and MySQL

HTTP (default port 80)

we'll use Apache

```
sudo apt-get install apache2
```

next, check that it is running (it should be on port 80):

```
sudo netstat -alpn | grep ' LISTEN '
...
tcp6  0   0 :::80   :::*            LISTEN   7865/apache2
...
```

try browsing to your web site using a web browser (e.g., Google Chrome)

```
http://localhost
```

your web site's files are in /var/www

```
sudo vim /var/www/index.html
```

Cyber Storm: tag is somewhere on the web page
so modify index.html appropriately

FTP (default port 21)

we'll use vsftpd (very secure FTP daemon)

```
sudo apt-get install vsftpd
```

next, check that it is running (it should be on port 21):

```
sudo netstat -alpn | grep ' LISTEN '
...
tcp   0   0 0.0.0.0:21   0.0.0.0:*   LISTEN   3309/vsftpd
...
```

try logging in using your user credentials:

```
$ ftp localhost
```

{you should be in your home directory – try `ls -lh`)
{Ctrl+D disconnects you}

let's modify the configuration to allow anonymous logins:

```
sudo vim /etc/vsftpd.conf

anonymous_enable=YES
local_enable=NO
ftpd_banner=Whatever you want
```

{of course, change this to something more appropriate}
{also, uncomment this line}

```
anon_root=***
```

{change *** to something appropriate}
{e.g., /home/ftp – you will need to create the directory with valid permissions}

restart the FTP server:

```
sudo service vsftpd restart
```

try logging in again (this time, anonymously):

```
ftp localhost
```

{user is anonymous}
{password is blank (just press Enter)}

Cyber Storm: tag is in the banner
so modify the banner appropriately

SSH (default port 22)

we'll use OpenSSH

```
sudo apt-get install openssh-server
```

next, check that it is running (it should be on port 22):

```
sudo netstat -alpn | grep ' LISTEN '
...
tcp   0   0 0.0.0.0:22    0.0.0.0:*    LISTEN   1091/sshd
...
```

try logging in using your user credentials:

```
ssh localhost
```

{you should be in your home directory – try `ls -lh`)
{Ctrl+D disconnects you}

where's the message of the day (MOTD)?

```
sudo vim /etc/motd
```

the file will most likely not exist

logging in as a different user (it must exist on your system) is easy:

```
ssh user@localhost
```

you can also disable password logins

why? well, you can, instead, use encryption keys
this is much more secure
see the separate document on the web site for this

make sure to add the following line to /etc/ssh/sshd_config
this is important during labs, challenges, and Cyber Storm

```
UseDNS no
```

Cyber Storm: tag is in the MOTD
so modify the MOTD appropriately

MySQL (default port 3306)

of course, we'll use MySQL!

```
sudo apt-get install mysql-server
```

this will also prompt you for a root password
you should probably make it a good one
you should probably remember it

let's secure the MySQL server

```
sudo mysql_secure_installation
```

I wouldn't worry about changing the root password
I would remove anonymous users
I would disable remote root logins
I would remove test databases
Yes, reload table privileges

next, check that it is running (it should be on port 3306):

```
sudo netstat -alpn | grep ' LISTEN '
...
tcp   0  0 127.0.0.1:3306  0.0.0.0:*  LISTEN  1389/mysqld
...
```

127.0.0.1?

yup, it means that it only accepts local (not remote) connections
i.e., it is listening on the localhost only

try logging in using root credentials:

```
mysql -uroot -p
```

so how do we accept remote connections?

first, let's add a new user (**to MySQL and not the entire server**)
we need to login to the MySQL server first

```
mysql -uroot -p
```

and now the new user

```
CREATE user 'dude'@'%' IDENTIFIED BY 'password';
```

% means that the user can connect from anywhere (including remotely)
this user can now be used to login and view databases
let's create a new database (as root)

```
CREATE DATABASE test;
```

now, let's login as the new user (Ctrl+D exits MySQL)

```
mysql -udude -p
```

and let's see the databases (test should be there)

```
SHOW DATABASES;
```

what about allowing remote access?

```
sudo vim /etc/mysql/my.cnf
```

comment out the bind-address option

```
#bind-address           = 127.0.0.1
```

restart the server

```
sudo service mysql restart
```

next, check that it is listening remotely:

```
sudo netstat -alpn | grep ' LISTEN '
...
tcp   0   0 0.0.0.0:3306   0.0.0.0:*  LISTEN  1091/mysqld
...
```

Cyber Storm: tag is in the name of a database
so create a database appropriately

changing ports

in most cases, the default port is specified in a configuration file
so you just need to change it

you will need to restart the appropriate server after changing the port
you can check that it is listening on the new port (via netstat)
of course, try logging in again on the new port

HTTP

```
sudo vim /etc/apache2/ports.conf
Listen 80
```

{you should change to a different port; e.g., 12345}

```
sudo vim /etc/apache2/sites-enabled/000-default.conf
```

change the following line to match the new port:

```
<VirtualHost *:12345>
```

restart the server

```
sudo service apache2 restart
```

browse to the new port in a web browser

```
http://localhost:12345
```

FTP

```
sudo vim /etc/vsftpd.conf
```

add the following line to match the new port:

```
listen_port=54321
```

restart the server

```
sudo service vsftpd restart
```

login on the new port

```
ftp localhost 54321
```

SSH

```
sudo vim /etc/ssh/sshd_config
Port 22
```

{you should change to a different port; e.g., 2222}

restart the server

```
sudo service ssh restart
```

login on the new port

```
ssh -p 2222 localhost
```

MySQL

```
sudo vim /etc/mysql/my.cnf
```

find the [client] section and change the port

```
port            = 3306
```

{you should change to a different port; e.g., 6033}

find the [mysqld] section and change the port

```
port            = 3306
```

{you should change to a different port; e.g., 6033}

restart the server

```
sudo service mysql restart
```

login on the new port

```
mysql -h {your IP} -udude -p
```