a. We run nmap to find out the open port on chadash.cs.columbia.edu

```
PORT     STATE    SERVICE
7/tcp    open     echo
13/tcp   open     daytime
21/tcp   open     ftp
22/tcp   open     ssh
23/tcp   open     telnet
80/tcp   open     http
111/tcp  open     rpcbind
135/tcp  filtered msrpc
1022/tcp open     unknown
3128/tcp open     squid-http
```

b. We can run nmap -A to get what protocol version are running at the ports

```
PORT     STATE    SERVICE    VERSION
7/tcp    open     echo
13/tcp   open     daytime
21/tcp   open     ftp        NetBSD ftpd
22/tcp   open     ssh         (protocol 2.0)
23/tcp   open     telnet     BSD-derived telnetd
80/tcp   open     http       Apache httpd 2.2.9 ((Unix) mod_ssl/2.2.9
OpenSSL/0.9.8e DAV/2)
111/tcp  open     rpcbind     2-4 (rpc #100000)
135/tcp  filtered msrpc
1022/tcp open     rpc
3128/tcp open     http-proxy Squid webproxy 2.6.STABLE21
```

c. The security issues with telnet and ftp are connected to the fact that these services do not use encryption and eavesdropping is possible both on the password as well as on the content transferred. Also ftp on chadash provides anonymous login, which can be an issue since anyone can login and this can lead to unwanted export of files (for example the "secret" file that yo can get from chadash using GET in ftp).

Rpcbind maps RPC program and version numbers to universal addresses. Checking with rpcinfo -p chadash.cs.columbia.edu we see that NFS is running on the machine. Using NFS makes implementing efficient firwall hard. Also there user authentication issues with NFS. It provides access based on IP addresses, which can be spoofed. The file /etc/export contains the IP addresses allowed to access the file system. This combined with the

fact that the anonymous ftp login provided access to file makes a big security issue (on chadash the file /etc/export does not exist).

Service such as echo and daytime can be used for DoS attacks. Daytime can give information about the time zone of machine.

If the Squid webproxy is not patched there is a vulnerability that may allow a use to poison the server caches by sending multiple Content-length headers in conjunction with specially crafted requests to poison the cache in certain situations. Squid is prone to a remote denial-of-service vulnerability caused by an unspecified error when processing requests with malformed HTTP version numbers.