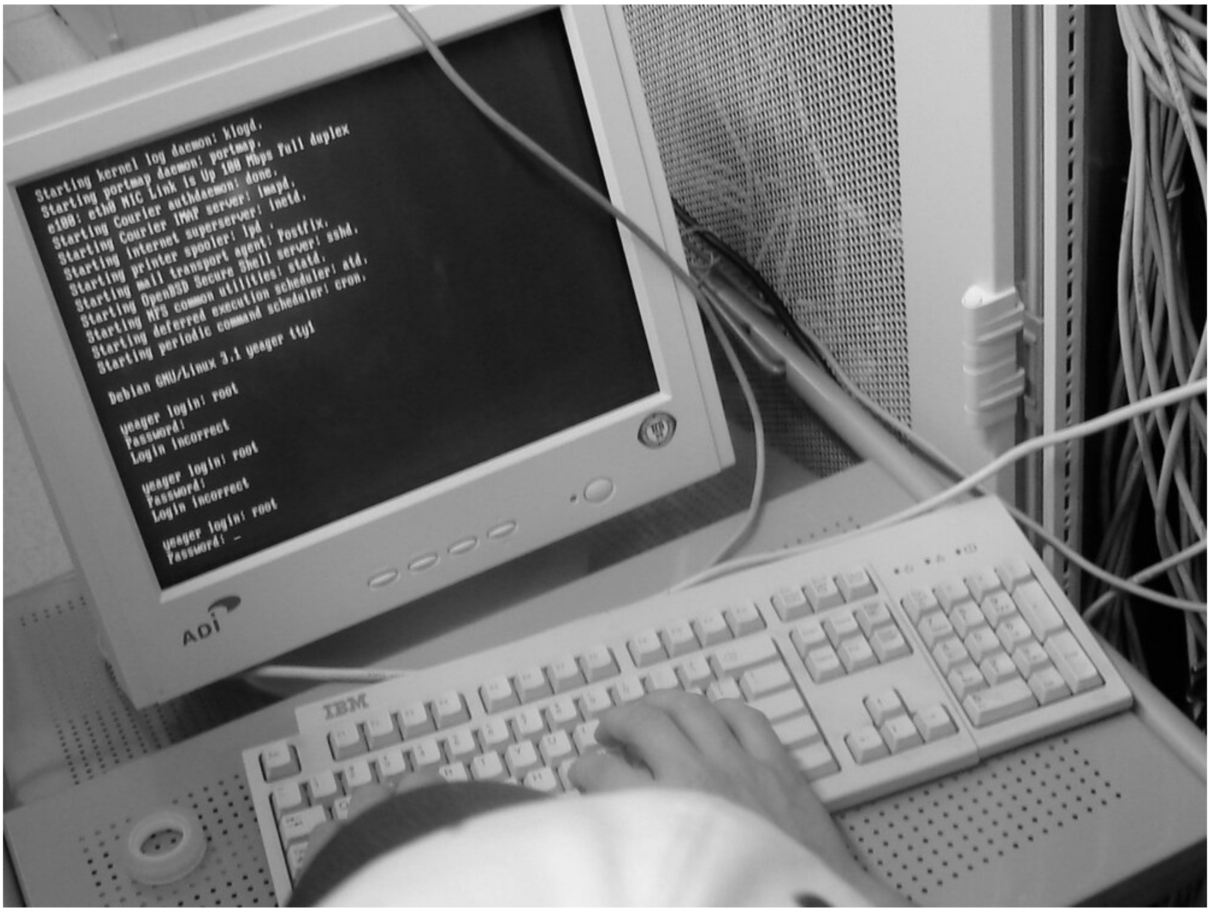
Cyber security: 5 easy tips to protect your server against hackers

Server hardening best practices for Windows and Linux

Willem L. Middelkoop

Mar. 10, 2018



```
Starting kernel log daemon: klogd.
Starting portmap daemon: portmap.
e100: eth0 NIC Link is Up 100 Mbps Full duplex
Starting Courier authdaemon: done.
Starting Courier IMAP server: imapd.
Starting internet superserver: inetd.
Starting printer spooler: lpd .
Starting mail transport agent: Postfix.
Starting OpenBSD Secure Shell server: sshd.
Starting NFS common utilities: statd.
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.

Debian GNU/Linux 3.1 yeager tty1

yeager login: root
Password:
Login incorrect

yeager login: root
Password:
Login incorrect

yeager login: root
Password: _
```

This week one of my clients was hacked and asked me for emergency assistance to help secure their server infrastructure. It was a web server that ran WordPress websites on Apache (with PHP/MySQL), including a few webshops with customer data. This hack could easily have been prevented with the following best practices, is *your* server secure?

The data centre is where your server lives. Even though it may be physically secure, you should check upon its software too!

1) Install less software

Cyber security is difficult enough, you should make it easier for yourself by installing less software. Fewer programs, services, plugins, mean less things to worry about. In cyber security terminology this is called reducing the attack vector.

Reduce your attack vector by:

- **starting with a minimal base system**: do not begin with a full blown and bloated operating system, start with as little as possible and keep track on the things you add
- **only install what you absolutely need**: install tools, plugins add-ons and programs that you really - really - need. Be hard and determined: less is more!
- **check dependencies of things you install**: If you install anything make sure you check the dependencies; software often requires other software (that you do not necessarily want)

2) Close all network ports, filter those you can't block

Firewalls are used to filter network traffic and are available as standard system software on most operating systems. Limit the openings hackers have to your server.

Firewall configuration should:

- **adopt a default policy of blocking:** Most operating systems allow everything by default. Turn this around and block everything except that kind of traffic you expect and need.
- **check inbound and outbound:** Filter incoming and outgoing network traffic. This makes it much harder for hackers to come in (and get out - in the unfortunate case of a successful hack).
- **filter open ports:** Secure open network ports by filtering traffic based on source (IP-address) and/or state, only allow traffic from where you expect it to come from.

3) Hide all version information

The software your server runs is versioned, often a number indicating the exact date when it was build. Hackers can use this version information to lookup known security problems, vulnerabilities, and weaknesses.

Stop helping hackers by removing version information from:

- **web servers:** Apache, NGINX, Microsoft Internet Information Services, etc. Check your server by analysing the HTTP headers.
- **mail servers:** Postfix, Exim, Dovecot, Sendmail, etc. Often these servers communicate their version in a ”hello banner”, shown directly after establishing a connection using SMTP, IMAP, POP3.
- **web languages:** PHP, .NET, Java, etc. Sometimes these frameworks and scripting languages add their own HTTP header (”x-powered-by”) with version info.
- **WordPress:** plugins, themes, forms, galleries, webshops, etc. Version info is often included in HTML-output or in filenames of CSS, JavaScript and images.
- **file servers:** FTP, SFTP, WebDav, etc. These servers communicate their version info in their greeting, shown directly upon connecting, often before authentication
- **SSH:** Did you know OpenSSH communicates operating system version info by default?

4) Use certificate/key authentication instead of passwords

If password-based logins are allowed, hackers can repeatedly attempt to access the server. With modern computing power it's easy to automate this guessing by trying combination after combination until the right password is found (brute forcing).

Secure authentication by:

- **use SSH key authentication:** an SSH key is much longer than a normal password and contains different characters than ordinary readable letters and numbers. This results in more possible combinations, making it exponentially more difficult for hackers to find the right key.

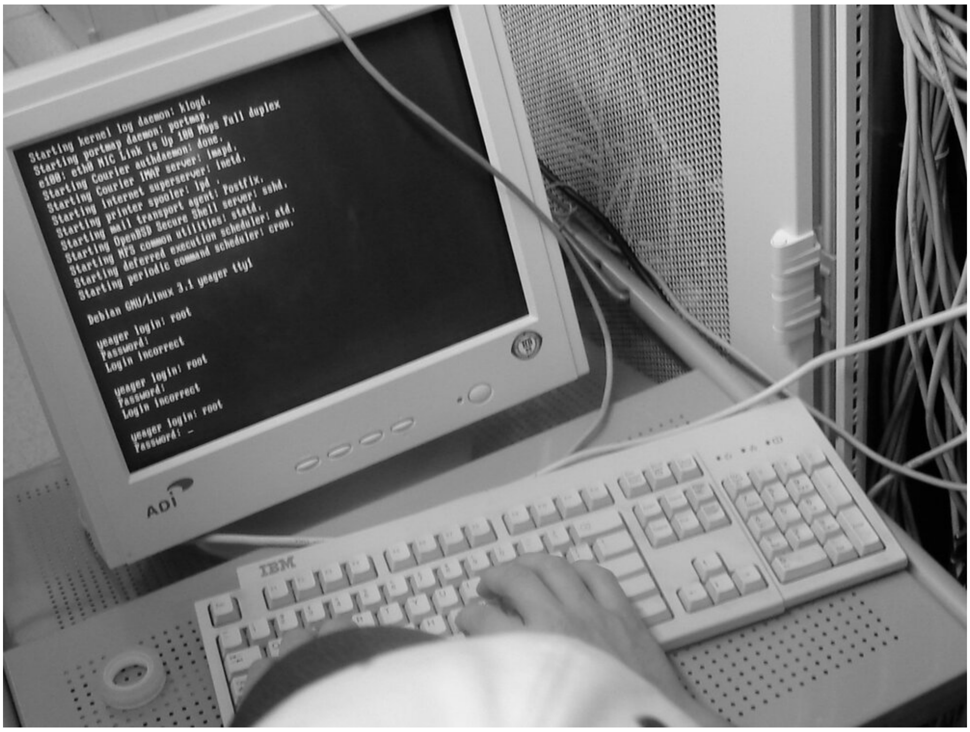
- **limit authentication rate**: Artificially make the password / key checking slower, reducing the speed of automated guessing
- **block automated guessing:** Exclude IP-addresses if they have failed to login successfully.

5) Check and update regularly

Most hacking is automated these days, bots are constantly scanning every server and website for exploitation opportunities. It's not a question *IF* they will find you, but *WHEN.*

Take care of your server by

- **checking its logs:** potential problems often become visible before any really bad things have happened. Check the server logs for errors and anomalies; often they're early signs of trouble.
- **check for updates:** either by using the software on your server or by checking the vendor / software website.
- **update regularly:** don't wait until it's too late, install updates as soon as possible (but *after* you've tested them!)



```
Starting kernel log daemon: klogd.
Starting portmap daemon: portmap.
e100: eth0 NIC Link is Up 100 Mbps full duplex
Starting Courier authdaemon: done.
Starting Courier IMAP server: imapd.
Starting internet superserver: inetd.
Starting printer spooler: lpd .
Starting mail transport agent: Postfix.
Starting OpenBSD Secure Shell server: sshd.
Starting NFS common utilities: statd.
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.

Debian GNU/Linux 3.1 yeager tty1

yeager login: root
Password:
Login incorrect

yeager login: root
Password:
Login incorrect

yeager login: root
Password: _
```

Check upon your server regulary - or find somebody that does this for you.

Conclusion

If you implement these measures you can greatly improve the cyber security of your server. Protecting your server means better safety for your business, your organisation and your customers' data.

No (sane) security consultant will offer you guarantees, given enough resources and determination, hacks will always be possible. Be prepared by making backups and encrypting your data.

Hopefully these tips will help you, if you need additional help you can find my contact information here reach or check out my cyber security services.