

SECURE SERVER

DAVOR · GUTTIEREZ @ LINUX · COM


ABOUT ME ...

My name is Davor Guttierrez

davor.guttierrez@linux.com



FOUR MYTHS ABOUT LINUX SECURITY

- Linux is invulnerable and virus-free.
 - Virus writers do not target Linux because it has a low market share.
 - Windows malware cannot run on Linux.
 - On Linux you install software from software repositories, which contain only trusted software.
- 

LINUX DESKTOP SECURITY

- The majority of new users are coming from Windows environments, where security focuses mostly on anti-virus software. To understand security on Linux, you must shift your thinking from this point of view.
 - If I install an anti-virus program I'll be fine.
 - Security through obscurity keeps me safe.
 - I can browse however I want to because malware on the web is mostly designed for Windows.
 - I don't need to use fancy browser add-ons when using public access wifi because I use Ubuntu.
 - I don't need a firewall because Ubuntu has no open ports by default.
 - Windows malware can not compromise Ubuntu.
 - Ubuntu is harder to exploit than Windows, Mac OSX, whatever else - and it's targeted less than those other operating systems as well.


LINUX SERVER SECURITY

My first 15 minutes on new server installation:


- Our servers are configured with two accounts: root and user accounts in IPA Server. The user accounts has sudo access via an arbitrarily long password and is the account that system administrators log into. Sysadmins log in with their public keys, not passwords, so administration is as simple as keeping the *authorized_keys* file up-to-date across servers. Root login over ssh is disabled, and the users can only log in from our office IP block.

ON VIRGIN SYSTEM


Ubuntu, CentOS or Oracle Linux – commands are similar

- passwd
 - apt-get update; apt-get upgrade or yum update
 - fail2ban installation
 - configuration for LDAP or SSSD
 - sudoers configuration in IPA Server
 - setup firewall (ufw, csf or apf) and SELinux or AppArmor
 - install logwatch and Zabbix Agent
 - install TSM
- 


BASIC SECURITY ON LAMP SERVER - UBUNTU

- Install and configure Firewall – ufw
 - SSH - Key based login, disable root login and change port
 - Apache SSL - Disable SSL v3 support
 - Protect su by limiting access only to admin group
 - Harden network with sysctl settings
 - Disable Open DNS Recursion and Remove Version Info - Bind9 DNS
 - Prevent IP Spoofing
 - Harden PHP for security
- 


BASIC SECURITY ON LAMP SERVER - UBUNTU

- Restrict Apache Information Leakage
 - Install and configure Apache application firewall - ModSecurity
 - Protect from DDOS (Denial of Service) attacks with ModEvasive
 - Scan logs and ban suspicious hosts - DenyHosts and Fail2Ban
 - Intrusion Detection - PSAD
 - Check for RootKits - RKHunter and CHKRootKit
 - Scan open Ports – Nmap, Lynis
 - Analyse system LOG files - LogWatch
 - SELinux - Apparmor
 - Audit your system security - Tiger
- 


INSTALL AND CONFIGURE FIREWALL

- install ufw
 - allow access to ssh and http
 - enable ufw
 - look at ufw status
- 

SSH HARDENING

- key based login
 - disable root login and
 - change SSH port
 - allowed users from allowed IP's
 - TCP Wrappers setup for different networks
- 

APACHE SSL HARDENING

- disable unsecure protocols
 - BEAST Attack
 - CRIME Attack
 - Heartbleed
 - FREAK Attack
 - Perfect Forward Secrecy
- 

DISABLE UNSECURE PROTOCOLS

SSLv2 and SSLv3

SSL v2 is insecure, so we need to disable it. We also disable SSLv3, as TLS 1.0 suffers a downgrade attack, allowing an attacker to force a connection to use SSLv3 and therefore disable forward secrecy.

SSLv3 allows exploiting of the POODLE bug.

Edit the config file:

```
SSLProtocol All -SSLv2 -SSLv3
```

All is a shortcut for +SSLv2 +SSLv3 +TLSv1 or - when using OpenSSL 1.0.1 and later - +SSLv2 +SSLv3 +TLSv1 +TLSv1.1 +TLSv1.2, respectively.

BEAST ATTACK

In short, by tampering with an encryption algorithm's CBC - cipher block chaining - mode's, portions of the encrypted traffic can be secretly decrypted.

Recent browser versions have enabled client side mitigation for the beast attack. The recommendation was to disable all TLS 1.0 ciphers and only offer RC4. However, [RC4 has a growing list of attacks against it],(<http://www.isg.rhul.ac.uk/tls/>) many of which have crossed the line from theoretical to practical. Moreover, there is reason to believe that the NSA has broken RC4, their so-called "big breakthrough."

Disabling RC4 has several ramifications. One, users with bad browsers such as Internet Explorer on Windows XP will use 3DES. Triple-DES is more secure than RC4, but it is significantly more expensive. Your server will pay the cost for these users. Two, RC4 mitigates BEAST. Thus, disabling RC4 makes TLS 1.0 users susceptible to that attack, by moving them to AES-CBC (the usual server-side BEAST "fix" is to prioritize RC4 above all else)

Indeed, with client-side mitigation (which Chrome and Firefox both provide), BEAST is a nonissue. But the risk from RC4 only grows: More cryptanalysis will surface over time.

CRIME ATTACK

The CRIME attack uses SSL Compression to do its magic, so we need to disable that. On Apache 2.2.24+ we can add the following line to the SSL config file we also edited above:

```
SSLCompression off
```

If you are using an earlier version of Apache and your distro has not backported this option then you need to recompile OpenSSL without ZLIB support. This will disable the use of OpenSSL using the DEFLATE compression method. If you do this then you can still use regular HTML DEFLATE compression.

HEARTBLEED

Heartbleed is a security bug disclosed in April 2014 in the OpenSSL cryptography library, which is a widely used implementation of the Transport Layer Security (TLS) protocol. Heartbleed may be exploited regardless of whether the party using a vulnerable OpenSSL instance for TLS is a server or a client. It results from improper input validation (due to a missing bounds check) in the implementation of the DTLS heartbeat extension (RFC6520), thus the bug's name derives from "heartbeat". The vulnerability is classified as a buffer over-read, a situation where more data can be read than should be allowed.

FREAK ATTACK

FREAK is a man-in-the-middle (MITM) vulnerability discovered by a group of cryptographers at

INRIA, Microsoft Research and IMDEA. FREAK stands for "Factoring RSA-EXPORT Keys."

The vulnerability dates back to the 1990s, when the US government banned selling crypto software overseas, unless it used export cipher suites which involved encryption keys no longer than 512-bits.

PERFECT FORWARD SECRECY

The concept of forward secrecy is simple: client and server negotiate a key that never hits the wire, and is destroyed at the end of the session. The RSA private from the server is used to sign a Diffie-Hellman key exchange between the client and the server. The pre-master key obtained from the Diffie-Hellman handshake is then used for encryption. Since the pre-master key is specific to a connection between a client and a server, and used only for a limited amount of time, it is called Ephemeral.


With Forward Secrecy, if an attacker gets a hold of the server's private key, it will not be able to decrypt past communications. The private key is only used to sign the DH handshake, which does not reveal the pre-master key. Diffie-Hellman ensures that the pre-master keys never leave the client and the server, and cannot be intercepted by a MITM.

Apache prior to version 2.4.7 and all versions of Nginx as of 1.4.4 rely on OpenSSL for input parameters to Diffie-Hellman (DH). Unfortunately, this means that Ephemeral Diffie-Hellman (DHE) will use OpenSSL's defaults, which include a 1024-bit key for the key-exchange. Since we're using a 2048-bit certificate, DHE clients will use a weaker key-exchange than non-ephemeral DH clients.

For Apache, there is no fix except to upgrade to 2.4.7 or later. With that version, Apache automatically selects a stronger key.

If you have Apache 2.4.8 or later and OpenSSL 1.0.2 or later, you can generate and specify your DH params file.

PROTECT SU BY LIMITING ACCESS

- `sudo groupadd admin`
 - `sudo usermod -a -G admin <YOUR ADMIN USERNAME>`
 - `sudo dpkg-statoverride --update --add root admin 4750 /bin/su`
 - or use pam libraries for su
- 

HARDEN NETWORK WITH SYSCTL SETTINGS

IP Spoofing protection

```
net.ipv4.conf.all.rp_filter = 1
```

```
net.ipv4.conf.default.rp_filter = 1
```

Ignore ICMP broadcast requests

```
net.ipv4.icmp_echo_ignore_broadcasts = 1
```

Disable source packet routing

```
net.ipv4.conf.all.accept_source_route = 0
```

```
net.ipv6.conf.all.accept_source_route = 0
```

```
net.ipv4.conf.default.accept_source_route = 0
```

```
net.ipv6.conf.default.accept_source_route = 0
```

Ignore send redirects

```
net.ipv4.conf.all.send_redirects = 0
```

```
net.ipv4.conf.default.send_redirects = 0
```

Block SYN attacks

net.ipv4.tcp_syncookies = 1

net.ipv4.tcp_max_syn_backlog = 2048

net.ipv4.tcp_synack_retries = 2

net.ipv4.tcp_syn_retries = 5

Log Martians

net.ipv4.conf.all.log_martians = 1

net.ipv4.icmp_ignore_bogus_error_responses = 1

Ignore ICMP redirects

net.ipv4.conf.all.accept_redirects = 0

net.ipv6.conf.all.accept_redirects = 0

net.ipv4.conf.default.accept_redirects = 0

net.ipv6.conf.default.accept_redirects = 0

Ignore Directed pings

net.ipv4.icmp_echo_ignore_all = 1

DISABLE OPEN DNS RECURSION AND REMOVE VERSION INFO


- recursion no;
- version "Not Disclosed";

PREVENT IP SPOOFING


- `order bind,hosts`
- `nospoof on`



HARDEN PHP FOR SECURITY

- `disable_functions = exec,system,shell_exec,passthru`
 - `register_globals = Off`
 - `expose_php = Off`
 - `display_errors = Off`
 - `track_errors = Off`
 - `html_errors = Off`
 - `magic_quotes_gpc = Off`
- 

RESTRICT APACHE INFORMATION LEAKAGE

- ServerTokens Prod
 - ServerSignature Off
 - TraceEnable Off
 - Header unset ETag
 - FileETag None
- 

WEB APPLICATION FIREWALL - MODSECURITY

- ModSecurity is a web application firewall for the Apache web server. In addition to providing logging capabilities, ModSecurity can monitor the HTTP traffic in real time in order to detect attacks. ModSecurity also operates as a web intrusion detection tool, allowing you to react to suspicious events that take place at your web systems.

PROTECT FROM DDOS ATTACKS - MODEVASIVE

- Mod Evasive is an evasive maneuvers module for Apache that provides evasive action in the event of an HTTP DoS attack or brute force attack. It is also designed to be a detection and network management tool, and can be easily configured to talk to ipchains, firewalls, routers, and more. mod_evasive presently reports abuse via email and syslog facilities.

SCAN LOGS AND BAN SUSPICIOUS HOSTS


- DenyHosts and Fail2Ban



SELINUX - APPARMOR

- National Security Agency (NSA) has taken Linux to the next level with the introduction of Security-Enhanced Linux (SELinux). SELinux takes the existing GNU/Linux operating system and extends it with kernel and user-space modifications to make it bullet-proof.

SECURITY SCANNER AND AUDITING

- nmap
 - Lynis
 - Kali Linux distribution
- 

THINGS I HAVEN'T COVERED

- Once you've hardened your server, you're advised to run some vulnerability scans and penetration tests against it in order to check that it's actually as invincible as you're now hoping it is. This is a topic which requires a post all of its own so I won't be covering it in any detail here, but a good starting point if you're not already familiar with it is the excellent Nmap security scanner.