Treball final de Grau en Enginyeria Informàtica

Avaluació de rendiment dels mòduls de seguretat SELinux i Apparmor en sistemes GNU/Linux

Autor: Roger Ortega Lluelles

Tutor: Joaquín López Sánchez-Montañés

Juny 2020

Context

La seguretat de la informació avui dia, ha esdevingut una de les principals preocupacions per a les empreses privades, governs, institucions públiques, etc... Mantenir els sistemes informàtics protegits, és clau en aquests sectors.

Sempre existeix un conflicte entre seguretat i funcionalitat. Es a dir, ¿què ha de prevaler més; garantir que els sistemes funcionin correctament, de forma ràpida i donant una bona experiència d'usuari, però sense seguretat, o bé, que ho facin de forma segura a costa de penalitzar el funcionament òptim dels sistemes?

L'objectiu d'aquest TFG és realitzar un estudi tècnic dels mòduls de seguretat per al kernel SELinux i AppArmor. Amb aquest estudi tècnic es pretén aprofundir en aquests mòduls de seguretat. Estudiar quin impacte suposen en la seva utilització en els sistemes, per tal de poder dir si és un component vàlid per tal de protegir la informació amb la que treballa l'usuari, una empresa o una institució.

Paraules clau: SELinux, AppArmor, rendiment, seguretat, kernel.

Information security today has become one of the main concerns for private companies, governments, public institutions, etc ... Maintaining protected computer systems is key in these sectors.

However, there is always a security and functionality conflict. That is, what should prevail more; ensure that the systems work properly, quickly and giving a good user experience, but without security, or that they do so safely at the expense of penalizing the optimal operation of the systems?

The purpose of this TFG is to conduct a technical study of the security modules for kernel, SELinux and AppArmor. This technical study aims to delve into these security modules. Study the impact they have on their use in systems, in order to be able to tell if it is a valid component in order to protect the information that the user, a company or an institution works with.

Keywords: SELinux, AppArmor, performance, security, kernel.

Índex de continguts

Context....3
1.Introducció....6
1.1.Context i justificació del treball....6
1.2.Objectius del treball....6
1.3.Metodologia....6
1.4.Planificació del treball....7
1.5.Productes obtinguts....7
1.6.Estudi econòmic....7
2.Conceptes de seguretat en sistemes operatius....10
2.1.Domini de seguretat....10
2.2.Model d'anelles....11
2.3.Mecanismes de control d'accés....11
3.Informació tècnica....12
3.1.Linux Security Module....12
3.2.SELinux....12
3.3.AppArmor....15
3.4.Amazon Web Services (AWS)....17
3.5.Phoronix Test Suite....17
3.6.OpenBenchmarking....17
4.Preparació de l'entorn....18
4.1.Desplegament dels sistemes en AWS....18
4.1.1.Creació de les instàncies a AWS....18
4.2.Configuració de SELinux en els sistemes....24
4.2.1.Instal·lació i configuració del mòdul SELinux a Debian....24
4.2.2.Configuració SELinux a Red Hat Enterprise Linux....27
4.3.Configuració de AppArmor en els sistemes....30
4.4.Instal·lació de Phoronix Test Suite....32
5.Selecció de les proves a executar....36
5.1.pts/apache....36
5.2.pts/fftw....36
5.3.pts/nginx....37
5.4.pts/openssl....37
5.5.pts/encode-mp3....37
5.6.pts/ramspeed-1.4.2....37
5.7.pts/postmark....37
5.8.pts/ipc-benchmark....37
5.9.pts/ctx-clock....37
5.10.pts/sysbench....37
6.Execució de les proves i resultats....38
6.1.pts/apache....38
Entorn monocore....38
Entorn Multicore....39
6.2.pts/fftw....40
Entorn monocore....40
Entorn multicore....40
6.3.pts/nginx....41
Entorn monocore....41
Entorn multicore....42
6.4.pts/openssl....43
Entorn monocore....43
Entorn multicore....43
6.5.pts/encode-mp3....44
Entorn monocore....44

Entorn multicore....45
6.6.pts/rampspeed-1.4.2....46
Entorn monocore....46
Entorn multicore....48
6.7.pts/postmark....50
Entorn monocore....50
Entorn multicore....51
6.8.pts/ipc-benchmark....52
Entorn monocore....52
Entorn multicore....53
6.9.pts/ctx-clock....55
Entorn monocore....55
Entorn multicore....56
6.10.pts/sysbench....57
Entorn monocore....57
Entorn multicore....58
7.Conclusions....59
Appendix....62
Bibliografia....65

1. Introducció

El present treball pretén realitzar un estudi tècnic dels mòduls de seguretat del nucli de sistemes operatius GNU/Linux encarregats d'implementar el control d'accés obligatori. S'analitzaran dues de les solucions més esteses en la majoria de les distribucions més conegudes, es realitzaran proves de rendiment per avaluar l'impacte d'aquests mòduls i finalment, s'extrauran conclusions al respecte.

1.1. Context i justificació del treball

En el context actual en que la seguretat de la informació ha esdevingut clau alhora de tractar les dades en sistemes informàtics, es fa necessari conèixer mecanismes que permetin implementar proteccions enfront a possibles atacs en els sistemes. Els mòduls de seguretat del nucli són uns components poc coneguts per la majoria d'administradors de sistemes GNU/Linux, inclús per alguns d'ells són mes aviat un problema.

Conèixer aquests mòduls, entendre el seu funcionament, veure que hi ha distribucions que ja proporcionen aquestes funcionalitats amb una instal·lació per defecte, esdevé necessari per començar a conèixer el seu potencial. Òbviament, s'ha de contrastar quin impacte tenen aquests proteccions sobre el rendiment dels sistemes. Tot això motiva la realització d'aquest treball final de grau.

1.2. Objectius del treball

- Presentar el conceptes de seguretat dels sistemes operatius.
- Aprofundir en el mecanisme de seguretat de control d'accés obligatori.
- Analitzar dues implementacions d'aquest mecanisme: SELinux i AppArmor.
- Estudiar l'impacte funcional d'aquests mecanismes en els sistemes operatius GNU/Linux.

1.3. Metodologia

Per realitzar aquest treball ha sigut necessari disposar de diversos sistemes amb diferents distribucions GNU/Linux, en concret, Debian, Red Hat Enteprise Linux i Ubuntu. En el sistema Debian s'ha instal·lat el sistema des de zero, en els altres s'ha aprofitat que ho porten incorporat com a part de la instal·lació per defecte. Finalment, s'han realitzar proves de rendiment mitjançant el framework Phoronix Test Suite i s'han analitzat els resultats per extreure les conclusions de l'estudi tècnic elaborat.

1.4. Planificació del treball

1. Conceptes de seguretat del sistemes operatius.
    - Dominis de seguretat
    - Mecanismes de control d'accés
2. Informació tècnica dels components de l'estudi tècnic
    - Linux Security Modules
    - SELinux
    - AppArmor
    - Amazon Web Services i EC2
    - Phoronix Test Suite
3. Configuració i instal·lació dels entorns de proves
    - Creació d'instàncies a AWS
    - Instal·lació i configuració de SELinux i AppArmor
    - Instal·lació i configuració de Phoronix Test Suite
4. Selecció i execució de les proves de rendiment
5. Conclusions

1.5. Productes obtinguts

Ja que el treball pretén realitzar un estudi tècnic no s'espera obtenir productes finals com a resultat, si no més aviat unes conclusions sobre la utilització d'aquests mecanismes de seguretat i del seu impacte sobre els sistemes operatius.

1.6. Estudi econòmic

Per a la execució del present treball, l'únic component de l'entorn que suposa un possible cost associat, és la utilització d'Amazon Web Services. Per tal de calcular els costos ens servirem de la calculadora d'Amazon Web Services que permet fer una estimació dels costos associats de la infraestructura necessària.

S'han utilitzat tres instàncies de tipus Linux per a l'entorn que s'ha anomenat monocore que correspondria a la configuració d'imatges de tipus t2.micro que tenen un cost associat unitari:

Operating system
Choose which operating system you'd like to run Amazon EC2 instances on.
Linux

Instance type
Search by name or enter the requirement to find the lowest cost instance for your needs.
Enter minimum requirements for each instance:
Search instances by name:

t2.micro

t2.micro

On-Demand hourly cost	vCPUs	GPUs
0.0116	1	NA
1YR Std reserved hourly cost	Memory (GiB)	Network performance
0.0072	1 GiB	Low to Moderate

Quantity
Enter the number of Amazon EC2 instances that you need.
3

Figura 1: Preu per hores de les instàncies t2.micro EC2 a AWS.

Pricing strategy Info

Pricing model
- EC2 Instance Savings Plans
- Compute Savings Plans
- Standard Reserved Instances
- Convertible Reserved Instances
- On-Demand Instances

Reservation term
- 1 Year
- 3 Year

Payment options
- No Upfront
- Partial Upfront
- Full Upfront

Show calculations

Unit conversions
EC2 Instance Savings Plans rate for t2.micro in the US East (Ohio) for 1 Year term and No Upfront is 0.0072 USD
Hours in the commitment: 365 days * 24 hours * 1 year = 8760.0000 hours
Total Commitment: 0.0072 USD * 8760 hours = 63.0720 USD
Upfront: No Upfront (0% of 63.072) = 0.0000 USD
Hourly cost for EC2 Instance Savings Plans = (Total Commitment - Upfront cost)/Hours in the term: (63.072 - 0.0000)/8760 = 0.0072 USD
*Please note that you will pay an hourly commitment for Savings Plans and your EC2 usage will be accrued at a discounted rate against this commitment.

Pricing calculations
3 instances x 0.0072 USD x 730 hours in month = 15.77 USD (monthly instance savings cost)
Amazon EC2 Instance Savings Plans instances (monthly): 15.77 USD

Figura 2: Estratègia de preus de les instàncies t2.micro EC2 a AWS.

Amazon Elastic Block Storage (EBS) Info
Attach persistent block storage volumes for your Amazon EC2 instances.

Storage per instance

Storage for each EC2 instance
General Purpose SSD (gp2)

Storage amount
10 GB

Show calculations

10 GB x 0.10 USD x 3 instances = 3.00 USD (EBS Storage Cost)
Amazon Elastic Block Storage (EBS) pricing (monthly): 3.00 USD

Figura 3: Preu del emmagatzemament a AWS de les instàncies EC2

Amazon EC2 estimate

Amazon EC2 Instance Savings Plans instances (monthly)	15.77 USD
Amazon Elastic Block Storage (EBS) pricing (monthly)	3.00 USD
Total monthly cost:	**18.77 USD**

Figura 4: Total del cost de les instàncies t2.micro EC2

De la mateixa manera tenim el cost associat a l'entorn multicore, que correspon a tres instàncies de tipus t2.xlarge amb una configuració en quant a emmagatzemament idèntic a l'escenari monocore:

Operating system
Choose which operating system you'd like to run Amazon EC2 instances on.
Linux

Instance type
Search by name or enter the requirement to find the lowest cost instance for your needs.
Enter minimum requirements for each instance:
Search instances by name:
t2.xlarge

t2.xlarge

On-Demand hourly cost	vCPUs	GPUs
0.1856	4	NA
1YR Std reserved hourly cost	Memory (GiB)	Network performance
0.115	16 GiB	Moderate

Quantity
Enter the number of Amazon EC2 instances that you need.
3

Figura 5: Preu per hores de les instàncies t2.xlarge EC2 a AWS.

Pricing strategy Info

Pricing model
- EC2 Instance Savings Plans
- Compute Savings Plans
- Standard Reserved Instances
- Convertible Reserved Instances
- On-Demand Instances

Reservation term
- 1 Year
- 3 Year

Payment options
- No Upfront
- Partial Upfront
- Full Upfront

Show calculations

Unit conversions
EC2 Instance Savings Plans rate for t2.xlarge in the US East (Ohio) for 1 Year term and No Upfront is 0.115 USD
Hours in the commitment: 365 days * 24 hours * 1 year = 8760.0000 hours
Total Commitment: 0.115 USD * 8760 hours = 1007.4000 USD
Upfront: No Upfront (0% of 1007.4) = 0.0000 USD
Hourly cost for EC2 Instance Savings Plans = (Total Commitment - Upfront cost)/Hours in the term: (1007.4 - 0.0000)/8760 = 0.1150 USD
*Please note that you will pay an hourly commitment for Savings Plans and your EC2 usage will be accrued at a discounted rate against this commitment.

Pricing calculations
3 instances x 0.115 USD x 730 hours in month = 251.85 USD (monthly instance savings cost)
Amazon EC2 Instance Savings Plans instances (monthly): 251.85 USD

Figura 6: Estratègia de preus de les instàncies t2.xlarge EC2 a AWS.

Amazon Elastic Block Storage (EBS) Info
Attach persistent block storage volumes for your Amazon EC2 instances.

Storage per instance

Storage for each EC2 instance
General Purpose SSD (gp2)

Storage amount
10 GB

Show calculations

10 GB x 0.10 USD x 3 instances = 3.00 USD (EBS Storage Cost)
Amazon Elastic Block Storage (EBS) pricing (monthly): 3.00 USD

Figura 7: Preu del emmagatzemament a AWS de les instàncies EC2

Amazon EC2 estimate

Amazon Elastic Block Storage (EBS) pricing (monthly)	3.00 USD
Amazon EC2 Instance Savings Plans instances (monthly)	251.85 USD
Total monthly cost:	**254.85 USD**

Figura 8: Total del cost de les instàncies t2.xlarge EC2

Amb aquestes estimacions s'opte un cost total de la infraestructura necessària per a dur a terme aquest TFG de 254,85 USD + 18,77 USD = 273,62 USD mensuals.

2. Conceptes de seguretat en sistemes operatius

En el mon de la informàtica quan es fa referència al concepte de seguretat, es fa referència a l'arquitectura de seguretat, on es parla de capes de protecció. Semblant al model OSI conceptual de xarxes, l'arquitectura de seguretat en sistemes engloba les següents capes:

1. Maquinari.
2. Kernel i controladors de dispositius.
3. Sistema operatiu.
4. Aplicacions.

2.1. Domini de seguretat

Un domini de seguretat defineix un conjunt d'objectes que una entitat te permís per accedir. En un sistema informàtic, això bé a definir els permisos d'accés que bé un usuari, un procés o una aplicació, tenen alhora d'accedir a la informació d'un sistema. En aquest sentit, existeixen dos dominis principals:

- Mode kernel: Sense cap tipus de restriccions. Utilitzat en les operacions bàsiques d'accés a memòria, disc, el que engloba l'activitat principal d'un sistema operatiu.
- Mode usuari: Destinat a totes les operacions que realitzarà un programa o comte d'usuari més ben dit. Aquesta separació permet que un error que es produeixi en un dels dos modes no afecti l'altre. Principalment els errors provocats per els usuaris.

2.2. Model d'anelles

El concepte d'anelles de seguretat defineix un determinat mode d'execució que tenen certes CPU, en el que es defineixen 4 anelles concèntriques, on la més interna és la que te més privilegis. Aquestes anelles són:

- Anella 0: Kernel.
- Anella 1: Altres components del sistema operatiu.
- Anella 2: Controladors de dispositius.
- Anella 3: Aplicacions d'usuari.

La comunicació entre aquestes anelles es duta a terme per les crides a sistema, això permet una abstracció als usuaris dels sistemes de les operacions importants i privilegiades en els sistemes.

Tot i aquest model teòric, a la pràctica suposaria un cost molt elevat d'operacions en el sistema i una efectivitat baixa.

2.3. Mecanismes de control d'accés

Aquest mecanisme de control d'accés d'usuaris als recursos del sistema esdevé clau alhora de proporcionar seguretat sobre el sistema. És important una adequada segregació de privilegis i una bona administració dels elements.

Existeixen tres tipus principals de mecanismes de control d'accés:

- *Control d'accés discrecional:* Aquest mecanisme està basat en la identitat del subjecte que accedeix al recurs. Normalment, el propietari de l'objecte defineix qui pot accedir al recurs. S'utilitzen atributs per tal d'establir els privilegis (lectura, escriptura, execució, etc...).
- *Control d'accés basat en rols:* Aquest sistema de control d'accés intenta aportar major granularitat que el discrecional. Es defineixen uns perfils o rols, i se'ls hi atribueix unes característiques que apliquen sobre el permisos i accions que poden fer. S'aconsegueix una bona segregació de permisos d'accés als recursos d'un sistema.
- *Control d'accés obligatori:* Aquest mecanisme es complementa amb els anteriors. Es basa en la definició d'unes polítiques d'accés per part d'un usuari "administrador". Aquestes polítiques són aplicades als objectes mitjançant etiquetes. Per tant, quan un usuari vol accedir a un objecte, es comprova la etiqueta de l'objecte per tal de veure si aquest usuari pot accedir-hi i quines accions pot realitzar

3. Informació tècnica

En la execució del present TFG s'han utilitzat els següents components tecnològics que tot seguit es descriuran per tal de contextualitzar l'escenari de l'estudi tècnic.

3.1. Linux Security Module

Són un conjunt d'extensions que es carreguen durant el procés d'arrancada del kernel dels sistemes operatius GNU/Linux. Tot i el nom no són realment mòduls carregables del sistema, si no més aviat components que es carreguen principalment a través de la opció de CONFIG_DEFAILT_SECURITY i que es poden escollir mitjançant el paràmetre "security=..." normalment introduït en el gestor d'arrancada del sistema operatiu.

Aquest tipus d'extensions són utilitzades principalment per implementar el mecanisme de control d'accés obligatori en sistemes GNU/Linux. També podrien ser utilitzats per implementar sistemes d'auditoria en el sistema operatiu a nivell de kernel.

Aquestes extensions són utilitzades per la majoria de projectes que implementen el mecanisme de control d'accés obligatori per a GNU/Linux com ara SELinux, AppArmor (les dues opcions que es tractaran en aquest TFG), Smack, Tomoyo, LoadPin,...

3.2. SELinux

Mòdul de seguretat per al kernel de Linux que implementa el mecanisme de control d'accés obligatori. És un mecanisme transparent per al usuari del sistema, està destinat a ser utilitzat i configurat únicament per l'administrador del sistema. Contràriament al sistema de control discrecional, on els usuaris propietaris, estableixen els permisos d'accés als objectes que ells defineixen.

Les autoritzacions o restriccions estan definides a les polítiques de SELinux, es a dir, en la política s'expliciten una serie de regles que s'apliquen sobre els contextos de seguretat. El context de seguretat, es defineix per tots els elements que estan supervisats per SELinux mitjançant etiquetes, que es representen com a atributs extres dels fitxers i directoris del sistema.

Aquests atributs o etiquetes es poden visualitzar afegint el paràmetre Z a la comanda ls -l i segueixen el següent format:

user:role:type:level(sensibilitat:categoria)

user: Identifica el usuari de SELinux, diferent al usuari de GNU/Linux.
role: Identifica el rol quan s'utilitza SELinux en mode *Role Based Access Control*.
type: Identifica el tipus de restricció. Les polítiques especifiquen bàsicament, com interactuen els tipus entre si mateixos.
level: Utilitzat en les polítiques avançades MLS/MCS. Aquestes permeten un control molt més precís i granular a través de la sensibilitat i la categoria. La primera és jeràrquica i la segona no.

Existeixen dos tipus de polítiques de SELinux:

- Targeted: Únicament determinats daemons del sistema GNU/Linux es troben supervisats per SELinux. És la opció per defecte que s'utilitza quan s'activa SELinux en un sistema, proporcionant

un control bastant òptim dels dimonis protegits. Aquests solen ser els bàsics del sistema com ara apache, dns, proxy squid, snmp i syslog.

- Multinivell o Multicategoria (MLS/MCS): Permet la implementació de polítiques avançades de seguretat mulitinivell i multicategoria on un sistema pot treballar amb diferents nivells de classificació de la informació.

SELinux pot treballar en tres modes:

- Enforcing: Aplica totes les polítiques definides per tant, denega o permet accessos/accions en el sistema.
- Permissive: Les polítiques són aplicades però no es denega o permeten les accions. Es registren totes les accions en un fitxer del sistema, podent identificar si s'ha violat o no una política.
- Disabled: En aquest mode SELinux es troba totalment desactivat. Ni tan sols es registren les accions que es realitzen.

Explicats tots aquests conceptes, es pot detallar com funciona SELinux en un sistema operatiu. Cada cop que el sistema operatiu ha d'executar una acció del sistema (ja sigui d'usuari, d'aplicació, procés, privilegiada, etc...) el kernel del sistema operatiu comprova si SELinux està activat en el sistema. Si no ho està, permet la execució de la operació.

En cas que SELinux estigui activat; el kernel del sistema operatiu registra l'acció que es vol dur a terme en el log de les accions de SELinux. Si el mode d'execució es permissive, permetrà la execució de l'acció. En cas que el mode de funcionament sigui enforcing, es comproven les regles definides a les polítiques per saber si l'acció que ha de ser executada està permesa o bé s'ha de denegar. Aquesta decisió be determinada per els contextos i les restriccions definides a les polítiques.

Amb els contextos de seguretat s'aconsegueix definir dominis sobre els que s'aplicaran les polítiques de SELinux. Trobem que per a tots els usuaris del sistema, existeix un context determinat, això permet afegir una capa de seguretat al sistema de control d'accés discrecional.

D'altra banda tenim que tots els processos i fitxers es troben etiquetats. Amb això aconseguim definir un domini d'execució dels processos fent que es trobin separats segons el context que tenen definit durant la etapa d'execució. Amb les polítiques es defineix la forma en que els processos hauran d'interactuar entre ells en funció del context que tinguin definit. D'aquesta forma s'està creant un confinament de processos en execució en el sistema.

Les avantatges que proporciona SELinux respecte un sistema on no s'implementa, principalment és que el administradors del sistema, poden limitar els permisos del sistema discrecional que els usuaris atorguen en el sistema. D'aquesta forma es limiten possibles forats de seguretat que es poden produir, ja sigui per usuaris o per la incorporació de programes en el sistema que no hagin estat programats correctament.

També pot servir de mecanisme de control per programes compromesos per un troià. A priori, limitaria les accions que un troià heretes del programa compromès. Per exemple, si un servidor web fos compromès per un atacant, si la política de SELinux està correctament configurada, estaria confinat en el context i domini del usuari, rol i tipus i el domini del procés. Així es denegaria l'accés de l'atacant a altres recursos. Sense SELinux, l'atacant podria accedir a tot allò que tingués permís segons el control d'accés discrecional.

3.3. AppArmor

Mecanisme de control d'accés obligatori en el que es confinen i limiten programes o un conjunt de recursos mitjançant Linux Security Modules. La filosofia de funcionament és la de limitar l'accés als programes i no als usuaris. Aquest confinament es fa a través de perfils carregats al kernel un cop s'arranca el sistema. D'aquesta manera es permet als usuaris administradors especificar aquelles activitats que es consideren legítimes per part del programes en els perfils de cada aplicació.

En els sistemes hi ha aplicacions més o menys exposades a possibles atacs. AppArmor permet als administradors crear perfils per aquelles aplicacions més susceptibles de ser atacades. Els usuaris no administradors, no són conscients de la existència d'aquest mecanisme alhora d'utilitzar les aplicacions del sistema. AppArmor proporciona una serie de perfils ja definits per a la protecció dels sistemes bàsics d'un sistema GNU/Linux.

AppArmor proporciona un control d'accés per als serveis de xarxa, especificant quins fitxers es permet llegir, escriure i executar per part de cada programa i a quin tipus de xarxa es pot accedir. Això garanteix que cada programa fa el que se suposa que ha de fer, i res més. Aquest confinament de programes permet a AppArmor protegir la resta del sistema de les accions no legítimes d'un possible procés compromès.

Modes de funcionament de AppArmor:

- Enforcement: En aquest mode, es carreguen els perfils al arrancar el sistema, aplicant la política definida al perfil, així com els informes d'intents de violació de la política (ja sigui mitjançant syslog o auditd).
- Complain: En aquest mode no s'aplicarà la política, sinó que únicament s'informa dels intents de violació d'aquesta, no hi ha cap restricció en les execucions realitzades per els programes en el sistema.
- Desactivat: En aquest mode de funcionament el mecanisme de control d'accés proporcionat per AppArmor es troba desactivat en el sistema i per tant no s'aplica cap protecció. Únicament, existeix el control d'accés discrecional tradicional de sistemes GNU/Linux.

Una de les diferències importants amb SELinux és que aquest identifica els objectes de sistema de fitxers pel nom de l'inode en lloc de fer-ho per la ruta com fa AppArmor. S'aclareix a continuació mitjançant el següent exemple disponible a la Wikipedia:

> *Hi ha dades inaccessibles sota AppArmor i SELinux i després un usuari crea una nova versió de la mateixa (una tècnica usada amb freqüència):*
>
> - *Sota SELinux les dades es tornen accessibles.*
> - *Sota AppArmor es continua negant l'accés.*
>
> *Hi ha dades inaccessibles sota AppArmor i SELinux i després un usuari crea un enllaç permanent:*
>
> - *En SELinux es continua negant l'accés.*
> - *En AppArmor les dades es tornen accessibles.*
>
> *En ambdós casos, una política per defecte de no permetre l'accés evitaria el problema.*

Una altra de les principals diferències, que potser aprecien més els administradors de sistema, és que AppArmor és més fàcil d'administrar que SELinux. La utilització dels perfils i la forma de crear-los i administrar-los és més fàcil que la definició de polítiques i contextos que proporciona SELinux. Tot i així, a nivell funcional els dos mecanismes proporcionen les mateixes proteccions.

3.4. Amazon Web Services (AWS)

Plataforma de computació en el núvol. Proporciona tots els nivells de computació en el núvol característiques d'aquest tipus d'entorns. IaaS (infraestructura en el núvol), PaaS (plataformes en el núvol), SaaS (Serveis o aplicacions en el núvol). En aquesta plataforma és on es desplegaran els sistemes per tal de realitzar les proves de rendiment per a avaluar l'impacte quan en el sistema està activat SELinux o AppArmor com a mecanismes de control d'accés, com a mòduls del sistema operatiu GNU/Linux.

Els sistemes seran desplegats mitjançant els components anomenats popularment com a EC2 a AWS. Això permet disposar de plantilles preconfigurades, per tal d'executar instàncies concretes d'aquestes plantilles a mode de màquina virtual. Podent escollir les característiques de número de CPU, memòria RAM, emmagatzemament, connectivitat de xarxa, etc...

3.5. Phoronix Test Suite

Eina que permet la execució d'un conjunt de proves de rendiment d'una forma automatitzada. De codi lliure i disponible en gran quantitat de sistemes operatius, Windows, GNU/Linux, Solaris, Mac OS X, BSD, etc...

Esdevé una de les plataformes més fiables a la hora de avaluar el rendiment d'un sistema en l'actualitat. Te una facilitat d'ús molt gran, així com un nombre molt elevat de proves a executar sobre els sistemes i la capacitat de processament automàtica dels resultats per presentar-los gràficament.

3.6. OpenBenchmarking

És una plataforma que serveix com a repositori dels resultats de les proves executades en els sistemes per avaluar el seu rendiment. El fet d'utilitzar aquest repositori de resultats, permet comparar les proves amb el resultat d'altres usuaris que hagin pogut executar les mateixes proves en d'altres sistemes. També permet disposar dels resultats en qualsevol moment, per accedir-hi on-line.

4. Preparació de l'entorn

4.1. Desplegament dels sistemes en AWS

S'han desplegat sis màquines EC2 a AWS per tal de configurar-hi els tres sistemes operatius on es duran a terme les proves de rendiment dels sistemes amb SELinux i AppArmor, activat i desactivat. S'ha optat per tres distribucions de GNU/Linux, Red Hat Enterprise Linux on el mòdul SELinux ja ve instal·lat com a part de la pròpia distribució, una màquina Debian on s'ha procedit a instal·lar i configurar SELinux des de zero i un sistema Ubuntu amb AppArmor ja instal·lat per defecte.

4.1.1. Creació de les instàncies a AWS

A la consola d'administració de AWS, s'accedeix a la secció de EC2 on es crearan les instàncies.

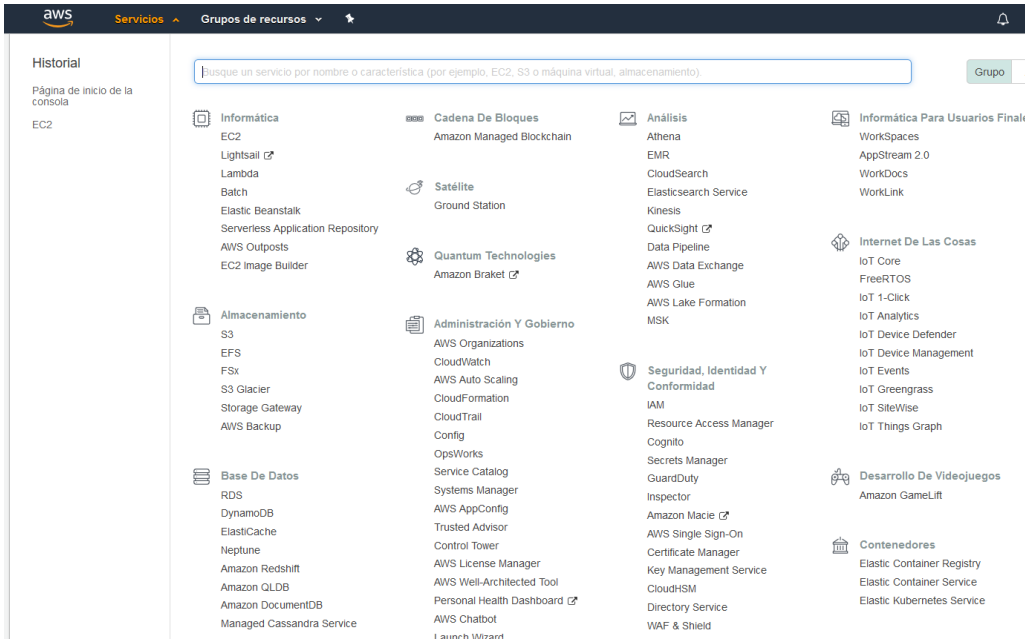


Figura 9: Serveis disponibles en un compte de AWS.

Tot seguit es procedeix al llançament de la instància, això desencadenarà un assistent que permet escollir els paràmetres bàsics de configuració per a aixecar la instància de la màquina.

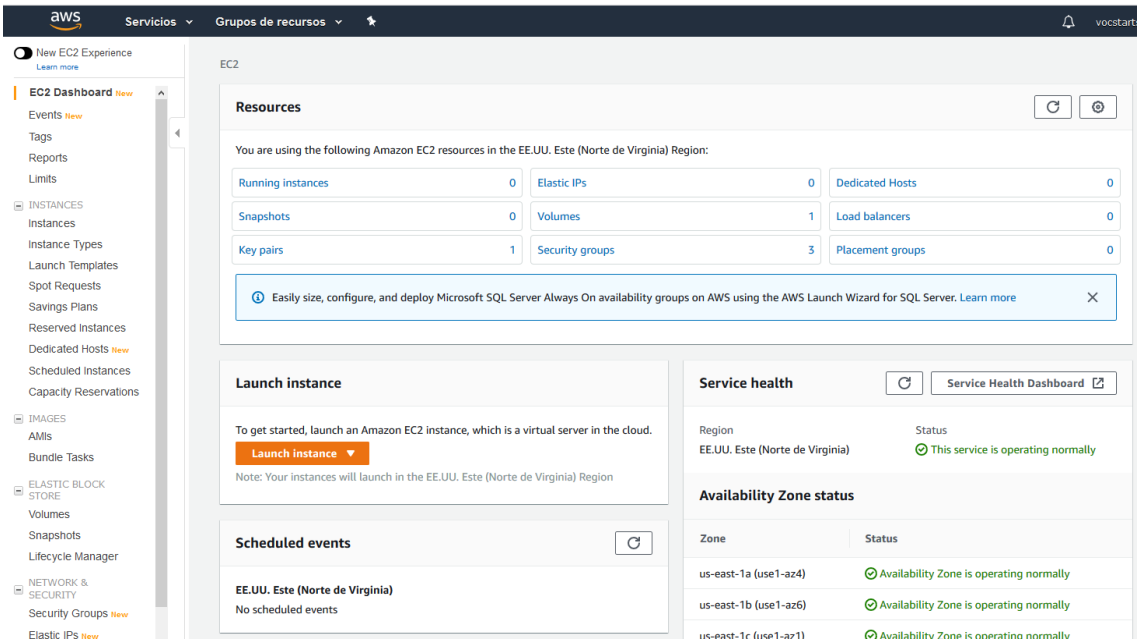


Figura 10: Opcions del servei EC2 de AWS.

Com a primer pas, s'ha d'escollir el tipus d'imatge de sistema operatiu que es vol configurar de la màquina:

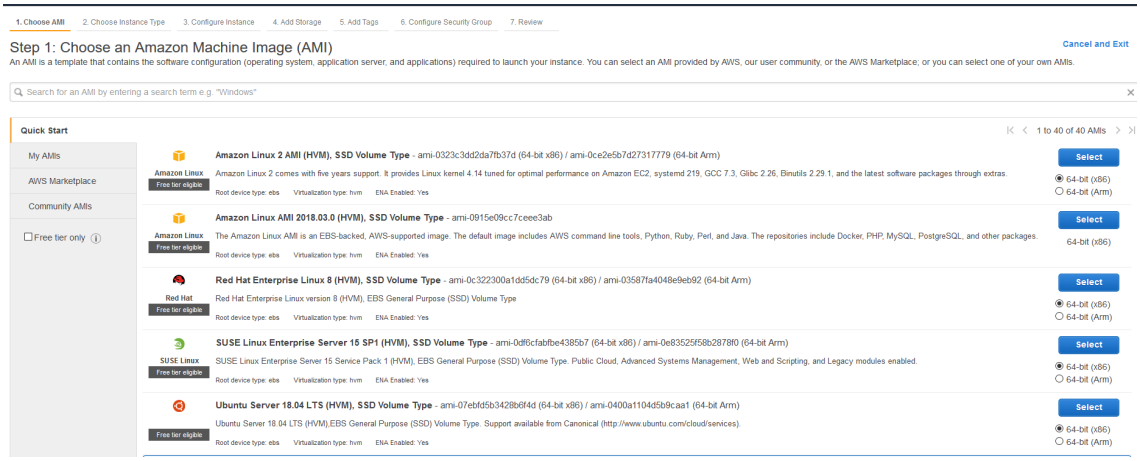


Figura 11: Selecció de la imatge de sistema operatiu per a la instància EC2.

Tot seguit s'escull el tipus d'instància, es a dir, la potència de la màquina. Les diferents instàncies de tipus t2 són les que es presenten en la següent taula:

Nombre	vCPU	RAM (GiB)
t2.nano	1	0,5
t2.micro	1	1,0
t2.small	1	2,0
t2.medium	2	4,0
t2.large	2	8,0
t2.xlarge	4	16,0
t2.2xlarge	8	32,0

Figura 12: Opcions de les configuracions de CPU i memòria de les instàncies t2.

Per a aquest TFG s'han escollit tres instàncies del tipus t2.micro, es a dir, amb 1 CPU virtual i amb 1 GiB de memòria. I també s'han escollit tres instàncies de tipus t2.xlarge, amb 4 CPU virtual i 16 GiB de memòria. Amb l'objectiu de diversificar els escenaris de les proves de rendiment.

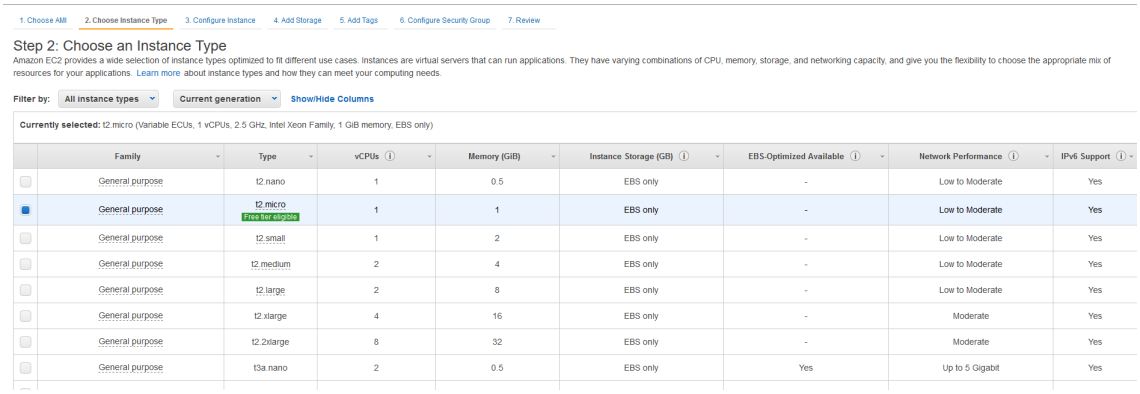


Figura 13: Selecció del tipus d'instància.

El següent paràmetre a escollir són detalls referents a la configuració de la xarxa o security group. Aquesta configuració s'ha deixat amb la que ve per defecte, sense modificar-la.

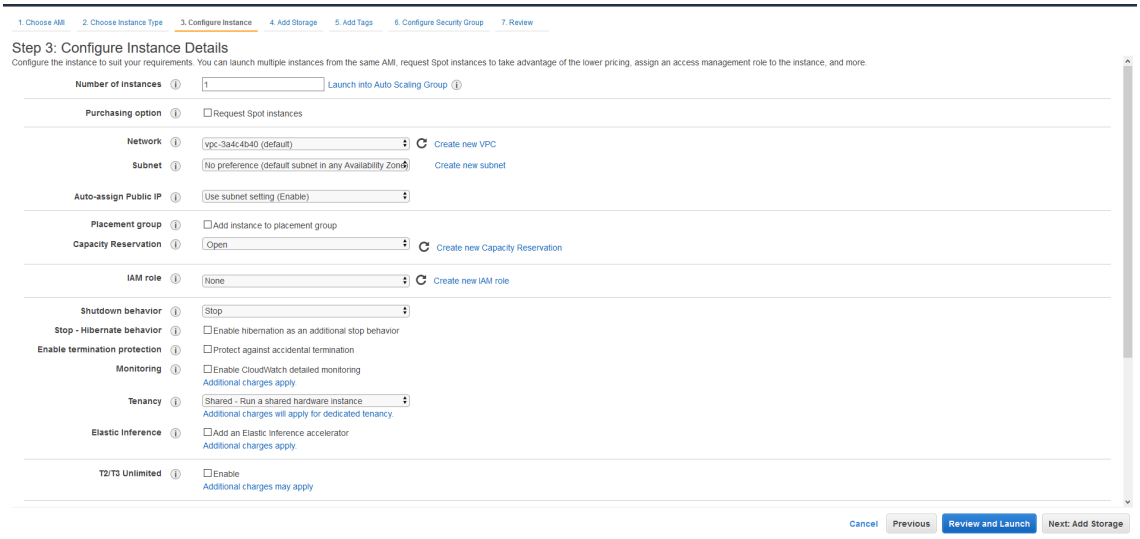


Figura 14: Selecció de les configuracions bàsiques de la instància.

El quart pas permet escollir el tipus disc dur que tindran les màquines. AWS permet escollir entre discos d'estat solid o discos magnètics. Per totes les màquines que s'han utilitzat alhora de fer les proves s'ha escollit discos de tipus gp2 (General Purpose SSD). Tot seguit es mostra els paràmetres d'aquests tipus de discos:

	Solid-state drives (SSD)	
		○ PostgreSQL ○ Oracle
API name	gp2	io1
Volume size	1 GiB - 16 TiB	4 GiB - 16 TiB
Max IOPS per volume	16,000 (16 KiB I/O) *	64,000 (16 KiB I/O) †
Max throughput per volume	250 MiB/s *	1,000 MiB/s †
Max IOPS per instance ††	80,000	80,000
Max throughput per instance ††	2,375 MB/s	2,375 MB/s
Dominant performance attribute	IOPS	IOPS

Figura 15: Opcions del tipus d'emmagatzemament disponible

La configuració escollida per a les màquines del present TFG ha estat:

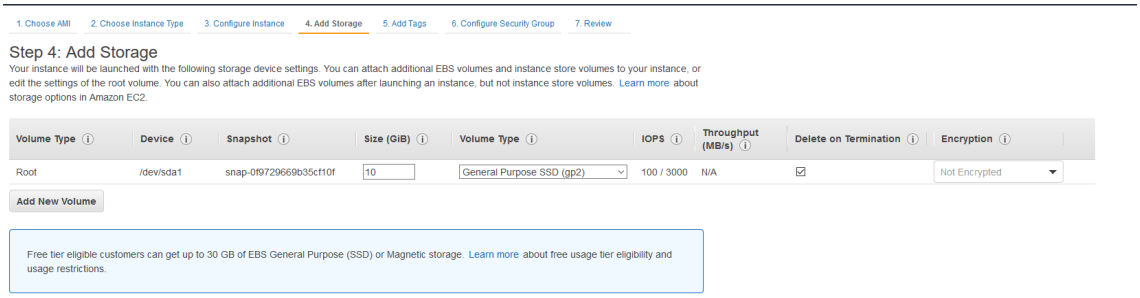


Figura 16: Selecció del tipus d'emmagatzemament.

El cinquè pas de la configuració permet assignar etiquetes a les màquines. Amb les etiquetes a AWS es permet realitzar certs tipus de configuracions avançades que per a dur a terme aquest TFG no són necessàries per tant no s'han aplicat etiquetes a cap de les màquines utilitzades.

El sisè pas és escollir la configuració del security group. Un security group a AWS, bàsicament seria un tallafoc a nivell de host, es a dir, quin tipus de connexions de xarxa es permet d'entrada o sortida a la màquina. Per a totes les màquines, s'ha permès tot el tràfic de sortida i només s'ha permès el tràfic d'accés per al port 22 ssh des de la IP NAT pública de la màquina.

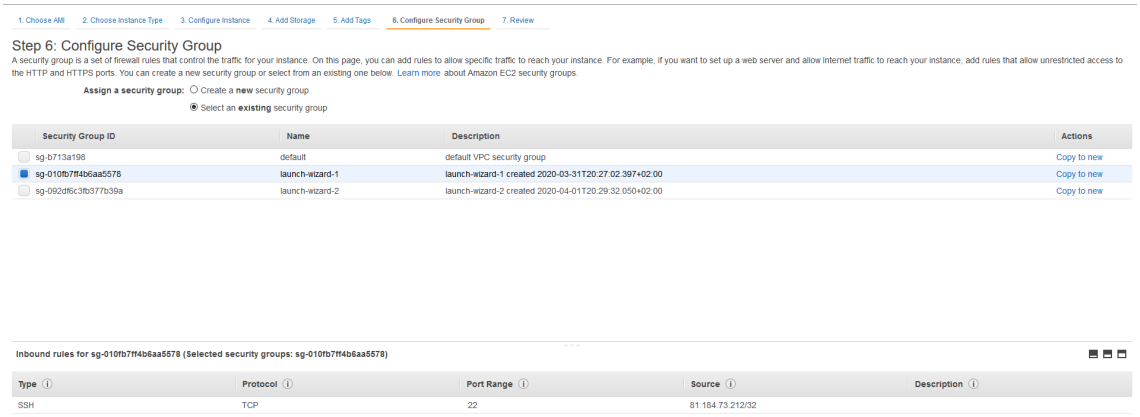


Figura 17: Selecció del tipus d'emmagatzemament.

Com a pas final, s'han de confirmar les configuracions escollides i fer l'arrancada de la màquina per tal que aquesta comenci a funcionar:

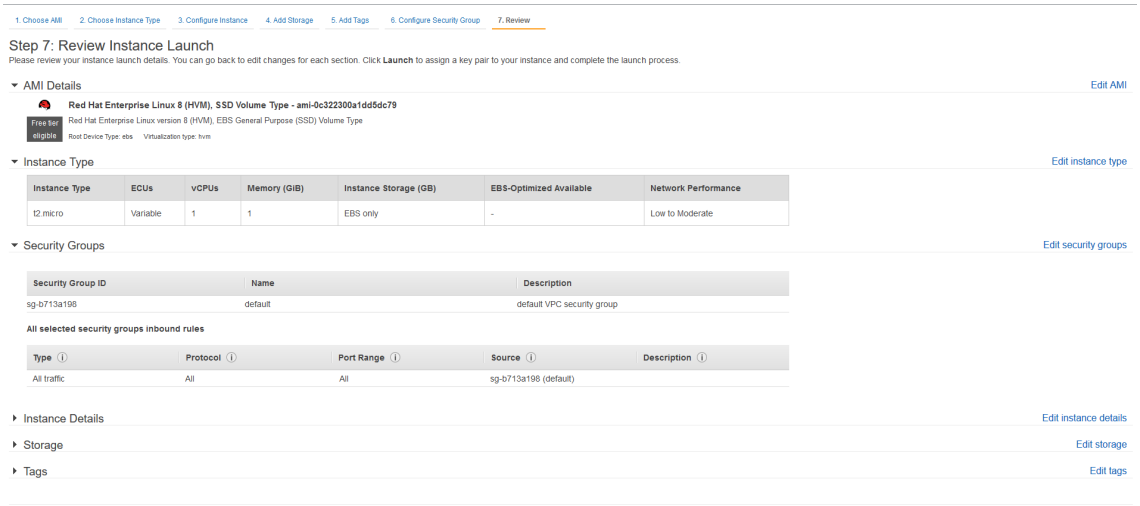


Figura 18: Resum de la configuració de la instància.

En aquest moment cal configurar el parell de claus públiques que s'utilitzarà per a l'accés remot a les màquines o bé, escollir un parell de claus ja existents. En aquesta ocasió, s'ha creat un parell de claus especific anomenat TFG:

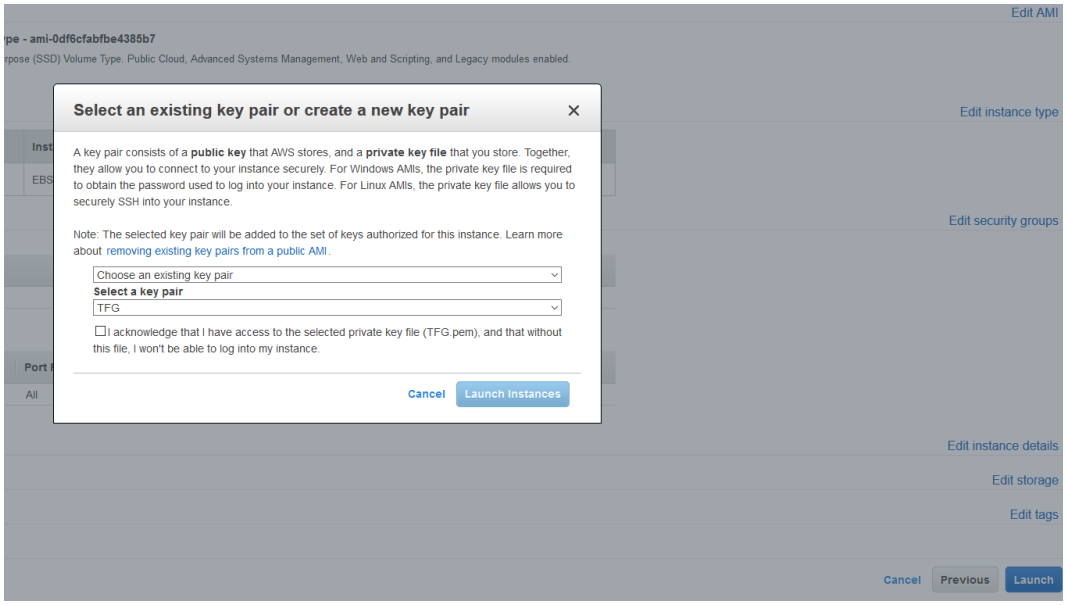
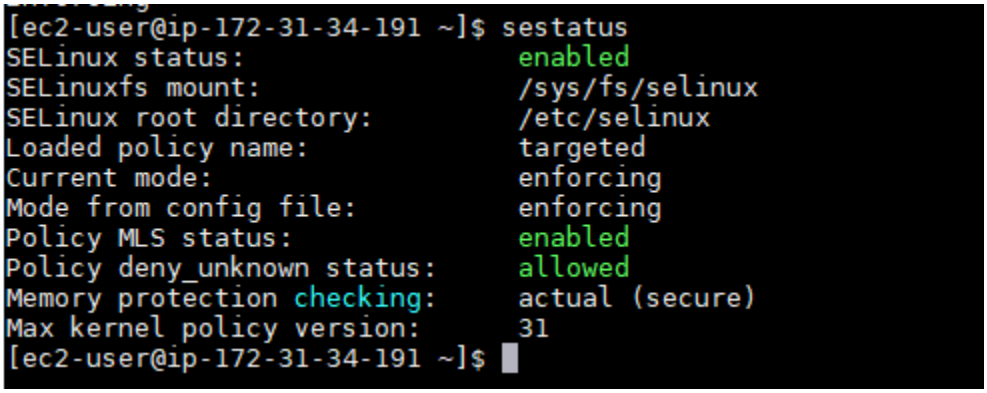


Figura 19: Selecció/creació de la clau pública per l'accés SSH.

4.2. Configuració de SELinux en els sistemes

En les distribucions Red Hat Enterprise Linux, el mòdul SELinux ja es troba instal·lat i configurat per defecte. Ho podem comprovar executant la comanda `sestatus`, d'on obtenim la següent resposta:



```
[ec2-user@ip-172-31-34-191 ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      31
[ec2-user@ip-172-31-34-191 ~]$
```

Figura 20: Comprovació de l'estat de SELinux a Red Hat Enterprise Linux.

S'evidencia que el mòdul SELinux està activat, en mode enforcing. SELinux un cop activat, te dos modes de funcionament. Mode permisive, les accions identificades com a prohibides en les polítiques queden registrades però executades en el sistema. El mode enforcing, si que ja prohibeix la execució de les accions prohibides a les polítiques.

Per al sistema Debian es necessari procedir a la instal·lació i configuració del mòdul SELinux ja que inicialment, no ve instal·lat per defecte. De fet, les últimes versions de Debian porten instal·lat i

configurat per defecte AppArmor. No és el cas de la versió utilitzada per a aquest TFG.

4.2.1. Instal·lació i configuració del mòdul SELinux a Debian

Com a primer pas es necessari comprovar si el mòdul AppArmor es troba instal·lat. Per la imatge que s'ha escollit, es veu clarament que no està present ni activat en el sistema. Alhora doncs, com a primer pas, també s'actualitzen els paquets del sistema operatiu.

```
Linux ip-172-31-83-157 4.9.0-8-amd64 #1 SMP Debian 4.9.130-2 (2018-10-27) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin@ip-172-31-83-157:~$ sudo systemctl stop apparmor
Failed to stop apparmor.service: Unit apparmor.service not loaded.
admin@ip-172-31-83-157:~$
admin@ip-172-31-83-157:~$ sudo apt update && sudo apt upgrade -yuf
Get:1 http://security.debian.org/debian-security stretch/updates InRelease [94.3 kB]
Ign:2 http://cdn-aws.deb.debian.org/debian stretch InRelease
Get:3 http://cdn-aws.deb.debian.org/debian stretch-updates InRelease [91.0 kB]
Get:4 http://cdn-aws.deb.debian.org/debian stretch Release [118 kB]
Get:5 http://cdn-aws.deb.debian.org/debian stretch Release.gpg [2,410 B]
Get:6 http://security.debian.org/debian-security stretch/updates/main amd64 Packages [521 kB]
Get:7 http://security.debian.org/debian-security stretch/updates/main Translation-en [231 kB]
Get:8 http://cdn-aws.deb.debian.org/debian stretch-updates/main amd64 Packages.diff/Index [13.0 kB]
Get:9 http://cdn-aws.deb.debian.org/debian stretch-updates/main Translation-en.diff/Index [7,132 B]
Get:10 http://cdn-aws.deb.debian.org/debian stretch-updates/main amd64 Packages [27.9 kB]
Get:11 http://cdn-aws.deb.debian.org/debian stretch-updates/main Translation-en [11.9 kB]
Get:12 http://cdn-aws.deb.debian.org/debian stretch/main amd64 Packages [7,083 kB]
Get:13 http://cdn-aws.deb.debian.org/debian stretch/main Translation-en [5,381 kB]
Fetched 13.6 MB in 3s (4,269 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
81 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
```

Figura 21: Comprovació de la presència de AppArmor en el sistema.

Arribats a aquest punt, es procedeix a la instal·lació del SELinux en el sistema. Per fer-ho s'executa la comanda `sudo apt install selinux-utils selinux-basics auditd audispd-plugins`. Tots aquests paquets són els necessaris en sistemes Debian GNU/Linux.

```
admin@ip-172-31-83-157:~$ sudo apt install selinux-utils selinux-basics auditd audispd-plugins
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils checkpolicy cpp cpp-6 g++ g++-6 gawk gcc gcc-6 libasan3 libatomic1 libauparse0 libblas-common libblas3 libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libgcc-6-dev libgfortran3
  libgomp1 libisl15 libitm1 liblapack3 liblsan0 libltdl7 libmpc3 libmpfr4 libmpx2 libperl5.24 libprelude2 libquadmath0 libsigsegv2 libstdc++-6-dev libtsan0 libubsan0 linux-libc-dev m4 make
  manpages-dev perl perl-modules-5.24 policycoreutils policycoreutils-dev policycoreutils-python-utils python3-audit python3-decorator python3-ipy python3-networkx python3-numpy
  python3-scipy python3-selinux python3-semanage python3-sepolgen python3-sepolicy python3-setools rename selinux-policy-default selinux-policy-dev setools
Suggested packages:
  binutils-doc cpp-doc gcc-6-locales g++-multilib g++-6-multilib gcc-6-doc libstdc++6-6-dbg gawk-doc gcc-multilib autoconf automake libtool flex bison gdb gcc-doc gcc-6-multilib libgcc1-dbg
  libgomp1-dbg libitm1-dbg libatomic1-dbg libasan3-dbg liblsan0-dbg libtsan0-dbg libubsan0-dbg libcilkrts5-dbg libmpx2-dbg libquadmath0-dbg glibc-doc libstdc++-6-doc m4-doc make-doc
  perl-doc libterm-readline-gnu-perl | libterm-readline-perl-perl python-networkx-doc gfortran python-numpy-doc python3-dev python3-nose python3-numpy-dbg python-scipy-doc logcheck
  syslog-summary setools-gui
The following NEW packages will be installed:
  audispd-plugins auditd binutils checkpolicy cpp cpp-6 g++ g++-6 gawk gcc gcc-6 libasan3 libatomic1 libauparse0 libblas-common libblas3 libc-dev-bin libc6-dev libcc1-0 libcilkrts5
  libgcc-6-dev libgfortran3 libgomp1 libisl15 libitm1 liblapack3 liblsan0 libltdl7 libmpc3 libmpfr4 libmpx2 libperl5.24 libprelude2 libquadmath0 libsigsegv2 libstdc++-6-dev libtsan0
  libubsan0 linux-libc-dev m4 make manpages-dev perl perl-modules-5.24 policycoreutils policycoreutils-dev policycoreutils-python-utils python3-audit python3-decorator python3-ipy
  python3-networkx python3-numpy python3-scipy python3-selinux python3-semanage python3-sepolgen python3-sepolicy python3-setools rename selinux-basics selinux-policy-default
  selinux-policy-dev selinux-utils setools
0 upgraded, 64 newly installed, 0 to remove and 0 not upgraded.
Need to get 64.5 MB of archives.
After this operation, 287 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Figura 22: Instal·lació de SELinux en un sistema Debian.

Acabada la instal·lació, comprovem el estat del mòdul en el sistema. Novament, executem la comanda `sestatus`.

```
admin@ip-172-31-83-157:~$ sudo sestatus
SELinux status:                 disabled
admin@ip-172-31-83-157:~$
```

Figura 23: Comprovació de l'estat de SELinux en un sistema Debian.

El resultat ens informa que el mòdul de seguretat ja es troba instal·lat en el sistema, però es troba desactivat.

Per tant, es necessari activar-lo per tal que es comencin a aplicar les polítiques sobre els objectes del sistema. Per a fer-ho s'executa la comanda `sudo selinux-activate` en el sistema.

```
admin@ip-172-31-83-157:~$ sudo selinux-activate
Activating SE Linux
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-4.9.0-12-amd64
Found initrd image: /boot/initrd.img-4.9.0-12-amd64
Found linux image: /boot/vmlinuz-4.9.0-8-amd64
Found initrd image: /boot/initrd.img-4.9.0-8-amd64
done
SE Linux is activated.  You may need to reboot now.
admin@ip-172-31-83-157:~$
```

Figura 24: Activació de SELinux en un sistema Debian.

Es necessari procedir a un reinici complert del sistema. Això és així atès que s'han d'etiquetar els objectes (fitxers) del sistema per tal de poder aplicar-hi les polítiques configurades d'accés a SELinux. Un cop reiniciat el sistema es comprova que SELinux està activat en mode permissive.

```
Linux ip-172-31-83-157 4.9.0-12-amd64 #1 SMP Debian 4.9.210-1 (2020-01-20) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Apr 22 18:09:07 2020 from 81.184.73.212
admin@ip-172-31-83-157:~$ sudo sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             default
Current mode:                   permissive
Mode from config file:          permissive
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Max kernel policy version:      30
admin@ip-172-31-83-157:~$
```

Figura 25: Comprovació de l'estat de SELinux en un sistema Debian amb resultat permissive.

Finalment, es procedeix a canviar al mode enforcing editant el fitxer /etc/selinux/config i modificant el paràmetre SELINUX a enforcing i reiniciant l'equip tot seguit:

```
admin@ip-172-31-83-157:~$ sudo sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             default
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Max kernel policy version:      30
admin@ip-172-31-83-157:~$
```

Figura 26: Comprovació de l'estat de SELinux en un sistema Debian amb resultat enabled.

4.2.2. Configuració SELinux a Red Hat Enterprise Linux

Arribats a aquest punt, podem realitzar algunes altres comprovacions en el sistema per inspeccionar la interacció de SELinux en el sistema. Si mirem el context del fitxer /etc/resolv.conf observem el següent resultat:

```
[ec2-user@ip-172-31-34-191 ~]$ ls -lZ /etc/resolv.conf
-rw-r--r--. 1 root root system_u:object_r:net_conf_t:s0 72 May  2 16:39 /etc/resolv.conf
```

Figura 27: Comprovació del context de SELinux d'un fitxer.

Podem observar com el paràmetre user en aquest cas és system_u, el rol és object_r el type és net_conf_t i te assignat el level S0 per si s'actives la política MLS/MCS. Ens està dient que aquest fitxer està restringits al usuari system i que és accessible per a operacions de xarxa per part del sistema. Te sentit, atès que és el fitxer on es poden configurar noms de servidors i les seves corresponents adreces IP.

Una altra comprovació que podem fer és llistar els usuaris SELinux que hi ha configurats en la política targeted (l'activada en el sistema). Per fer-ho executem la comanda `seinfo -u`:

```
[ec2-user@ip-172-31-34-191 ~]$ seinfo -u

Users: 8
   guest_u
   root
   staff_u
   sysadm_u
   system_u
   unconfined_u
   user_u
   xguest_u
[ec2-user@ip-172-31-34-191 ~]$
```

Figura 28: Llistat d'usuaris de SELinux.

S'observen vuit tipus d'usuaris SELinux entre ells el unconfined_u que és el que s'utilitza per a aquells objectes no confinats.

Per veure els rols que hi ha definits a la política, s'utilitza la comanda `seinfo -r`:

```
[ec2-user@ip-172-31-34-191 ~]$ seinfo -r

Roles: 14
   auditadm_r
   dbadm_r
   guest_r
   logadm_r
   nx_server_r
   object_r
   secadm_r
   staff_r
   sysadm_r
   system_r
   unconfined_r
   user_r
   webadm_r
   xguest_r
[ec2-user@ip-172-31-34-191 ~]$
```

Figura 29: Llistat de rols de SELinux.

En aquesta ocasió veiem que hi ha 14 rols definits a la política.

Tot seguit extraiem tota la informació de la política tal com es pot veure a la següent captura de pantalla:

```
[ec2-user@ip-172-31-34-191 ~]$ seinfo -x /etc/selinux/targeted/policy/policy.31
Statistics for policy file: /etc/selinux/targeted/policy/policy.31
Policy Version:             31 (MLS enabled)
Target Policy:              selinux
Handle unknown classes:     allow
  Classes:             131    Permissions:         457
  Sensitivities:         1    Categories:         1024
  Types:              4943    Attributes:          251
  Users:                 8    Roles:                14
  Booleans:            328    Cond. Expr.:         374
  Allow:            112512    Neverallow:            0
  Auditallow:          161    Dontaudit:         10278
  Type_trans:       241253    Type_change:          87
  Type_member:          35    Range_trans:        6015
  Role allow:           37    Role_trans:          422
  Constraints:          72    Validatetrans:         0
  MLS Constrain:        72    MLS Val. Tran:         0
  Permissives:           0    Polcap:                5
  Defaults:              7    Typebounds:            0
  Allowxperm:            0    Neverallowxperm:       0
  Auditallowxperm:       0    Dontauditxperm:        0
  Initial SIDs:         27    Fs_use:               33
  Genfscon:            106    Portcon:             640
  Netifcon:              0    Nodecon:               0
[ec2-user@ip-172-31-34-191 ~]$
```

Figura 30: Informació de la política de SELinux.

Com a última comprovació de la configuració de SELinux present al sistema, visualitzarem les accions que estan permeses en un dels tipus definits. En concret, el tipus httpd_log_t. Per a fer-ho executem la comanda `semanage fcontext -l | grep httpd_log_t`:

```
[ec2-user@ip-172-31-34-191 ~]$ semanage fcontext -l | grep httpd_log_t
ValueError: SELinux policy is not managed or store cannot be accessed.
[ec2-user@ip-172-31-34-191 ~]$ sudo semanage fcontext -l | grep httpd_log_t
/etc/httpd/logs                                    all files          system_u:object_r:httpd_log_t:s0
/srv/([^/]*/)?www/logs(/.*)?                       all files          system_u:object_r:httpd_log_t:s0
/var/lib/openshift/\.log/httpd(/.*)?               all files          system_u:object_r:httpd_log_t:s0
/var/log/apache(2)?(/.*)?                          all files          system_u:object_r:httpd_log_t:s0
/var/log/apache-ssl(2)?(/.*)?                      all files          system_u:object_r:httpd_log_t:s0
/var/log/cacti(/.*)?                               all files          system_u:object_r:httpd_log_t:s0
/var/log/cgiwrap\.log.*                            regular file       system_u:object_r:httpd_log_t:s0
/var/log/cherokee(/.*)?                            all files          system_u:object_r:httpd_log_t:s0
/var/log/dirsrv/admin-serv(/.*)?                   all files          system_u:object_r:httpd_log_t:s0
/var/log/glpi(/.*)?                                all files          system_u:object_r:httpd_log_t:s0
/var/log/graphite-web(/.*)?                        all files          system_u:object_r:httpd_log_t:s0
/var/log/horizon(/.*)?                             all files          system_u:object_r:httpd_log_t:s0
/var/log/httpd(/.*)?                               all files          system_u:object_r:httpd_log_t:s0
/var/log/lighttpd(/.*)?                            all files          system_u:object_r:httpd_log_t:s0
/var/log/nginx(/.*)?                               all files          system_u:object_r:httpd_log_t:s0
/var/log/php-fpm(/.*)?                             all files          system_u:object_r:httpd_log_t:s0
/var/log/php_errors\.log.*                         regular file       system_u:object_r:httpd_log_t:s0
/var/log/roundcubemail(/.*)?                       all files          system_u:object_r:httpd_log_t:s0
/var/log/suphp\.log.*                              regular file       system_u:object_r:httpd_log_t:s0
/var/log/thttpd\.log.*                             regular file       system_u:object_r:httpd_log_t:s0
/var/opt/rh/rh-nginx18/log(/.*)?                   all files          system_u:object_r:httpd_log_t:s0
/var/www(/.*)?/logs(/.*)?                          all files          system_u:object_r:httpd_log_t:s0
/var/www/miq/vmdb/log(/.*)?                        all files          system_u:object_r:httpd_log_t:s0
/var/www/openshift/broker/httpd/logs(/.*)?         all files          system_u:object_r:httpd_log_t:s0
/var/www/openshift/console/httpd/logs(/.*)?        all files          system_u:object_r:httpd_log_t:s0
/var/www/openshift/console/log(/.*)?               all files          system_u:object_r:httpd_log_t:s0
/var/www/stickshift/[^/]*/log(/.*)?                all files          system_u:object_r:httpd_log_t:s0
[ec2-user@ip-172-31-34-191 ~]$
```

Figura 31: Informació de les accions definides per httpd_log_t.

Observem sobre quins fitxers del sistema s'ha definit el tipus httpd_log_t, sobre quin usuari SELinux i més informació complementaria.

4.3. Configuració de AppArmor en els sistemes

En els sistemes Ubuntu la funcionalitat de AppArmor ja ve instal·lada i configurada per defecte. Tot seguit comprovem l'estat i la configuració actual dels perfils que hi ha en el sistema. Per tal de verificar que AppArmor s'està executant en el sistema, cal executar la comanda `aa-status`:

```
ubuntu@ip-172-31-85-159:~$ sudo aa-status
apparmor module is loaded.
22 profiles are loaded.
20 profiles are in enforce mode.
   /sbin/dhclient
   /snap/core/8935/usr/lib/snapd/snap-confine
   /snap/core/8935/usr/lib/snapd/snap-confine//mount-namespace-capture-helper
   /usr/bin/lxc-start
   /usr/bin/man
   /usr/lib/NetworkManager/nm-dhcp-client.action
   /usr/lib/NetworkManager/nm-dhcp-helper
   /usr/lib/connman/scripts/dhclient-script
   /usr/lib/snapd/snap-confine
   /usr/lib/snapd/snap-confine//mount-namespace-capture-helper
   /usr/sbin/tcpdump
   lxc-container-default
   lxc-container-default-cgns
   lxc-container-default-with-mounting
   lxc-container-default-with-nesting
   man_filter
   man_groff
   snap-update-ns.amazon-ssm-agent
   snap-update-ns.core
   snap.core.hook.configure
2 profiles are in complain mode.
   snap.amazon-ssm-agent.amazon-ssm-agent
   snap.amazon-ssm-agent.ssm-cli
1 processes have profiles defined.
0 processes are in enforce mode.
1 processes are in complain mode.
   snap.amazon-ssm-agent.amazon-ssm-agent (1336)
0 processes are unconfined but have a profile defined.
```

Figura 32: Comprovació de l'estat d'AppArmor.

Amb la sortida de la comanda s'aprecia clarament que; AppArmor està carregat en el sistema amb 22 perfils, d'aquests n'hi ha 20 que estan en mode enforce i 2 en mode complain.

Els principals fitxers de definició de les polítiques de AppArmor es troben a la ruta de sistema /etc/apparmor.d/. Executant un `ls -l` en el sistema Ubuntu s'observa el següent resultat:

```
ubuntu@ip-172-31-85-159:~$ ls -l /etc/apparmor.d/
total 76
drwxr-xr-x 5 root root  4096 Apr  8 17:38 abstractions
drwxr-xr-x 2 root root  4096 May  2 17:35 cache
drwxr-xr-x 2 root root  4096 Apr  8 17:36 disable
drwxr-xr-x 2 root root  4096 Apr 24  2018 force-complain
drwxr-xr-x 2 root root  4096 Apr  8 17:39 local
drwxr-xr-x 2 root root  4096 Apr  8 17:39 lxc
-rw-r--r-- 1 root root   198 Nov 23  2018 lxc-containers
-rw-r--r-- 1 root root  3194 Mar 26  2018 sbin.dhclient
drwxr-xr-x 5 root root  4096 Apr  8 17:38 tunables
-rw-r--r-- 1 root root   125 Nov 23  2018 usr.bin.lxc-start
-rw-r--r-- 1 root root  2857 Apr  7  2018 usr.bin.man
-rw-r--r-- 1 root root 23936 Oct 30  2019 usr.lib.snapd.snap-confine.real
-rw-r--r-- 1 root root  1550 Apr 24  2018 usr.sbin.rsyslogd
-rw-r--r-- 1 root root  1353 Mar 31  2018 usr.sbin.tcpdump
```

Figura 33: Principals fitxers d'AppArmor.

Al inspeccionar un dels fitxers, es pot apreciar les definicions de les accions que hi ha permeses en aquest perfil. Per a fer-ho, serà suficient en llistar el contingut del fitxer en qüestió. A la imatge apreciem la definició del perfil de rsyslog:

```
ubuntu@ip-172-31-85-159:~$ cat /etc/apparmor.d/usr.sbin.rsyslogd
# Last Modified: Sun Sep 25 08:58:35 2011
#include <tunables/global>

# Debugging the syslogger can be difficult if it can't write to the file
# that the kernel is logging denials to. In these cases, you can do the
# following:
# watch -n 1 'dmesg | tail -5'

/usr/sbin/rsyslogd {
  #include <abstractions/base>
  #include <abstractions/nameservice>

  capability sys_tty_config,
  capability dac_override,
  capability dac_read_search,
  capability setuid,
  capability setgid,
  capability sys_nice,
  capability syslog,

  unix (receive) type=dgram,
  unix (receive) type=stream,

  # rsyslog configuration
  /etc/rsyslog.conf r,
  /etc/rsyslog.d/ r,
  /etc/rsyslog.d/** r,
  /{,var/}run/rsyslogd.pid{,.tmp} rwk,
  /var/spool/rsyslog/ r,
  /var/spool/rsyslog/** rwk,

  /usr/lib{,32,64}/{,@{multiarch}/}rsyslog/*.so mr,

  /dev/tty*                     rw,
  /dev/xconsole                 rw,
  @{PROC}/kmsg                  r,

  /dev/log                      rwl,
  /{,var/}run/utmp              rk,
  /var/lib/*/dev/log            rwl,
  /var/spool/postfix/dev/log    rwl,
  /{,var/}run/systemd/notify    w,

  # 'r' is needed when using imfile
  /var/log/**                   rw,

  # Add these for mysql support
  #/etc/mysql/my.cnf r,
  #/{,var/}run/mysqld/mysqld.sock rw,

  # Add thes for postgresql support
  ##include <abstractions/openssl>
  ##include <abstractions/ssl_certs>
  #/{,var/}run/postgresql/.s.PGSQL.*[0-9] rw,

  # Site-specific additions and overrides. See local/README for details.
  #include <local/usr.sbin.rsyslogd>
}
ubuntu@ip-172-31-85-159:~$
```

Figura 34: Perfil definit per rsyslog a AppArmor.

4.4. Instal·lació de Phoronix Test Suite

És necessari preparar els sistemes per a la correcta execució de la suite Phoronix. A tal efecte s'han instal·lat els següents paquets:

Sistemes RedHat:

```
sudo yum groupinstall "Development Tools" -y
sudo yum install {libaio,pcre,popt}-devel glibc-{devel,static}
```

Sistemes Debian/Ubuntu:

```
sudo apt-get install build-essential
sudo apt-get install libpcre3 libpcre3-dev
sudo apt-get install libaio-dev
sudo apt-get install libpopt-dev
```

Per tal d'instal·lar la suite Phoronix a les màquines, es procedix a la descarrega i instal·lació dels comprimits *.tar.gz de la web del producte. Per a què aquest procediment sigui efectiu com a pas previ és necessari instal·lar els següents paquets en els sistemes: wget, php-cli php-xml, bzip2:

```
sudo yum install wget php-cli php-xml bzip2
sudo apt-get install wget php-cli php-xml bzip2
```

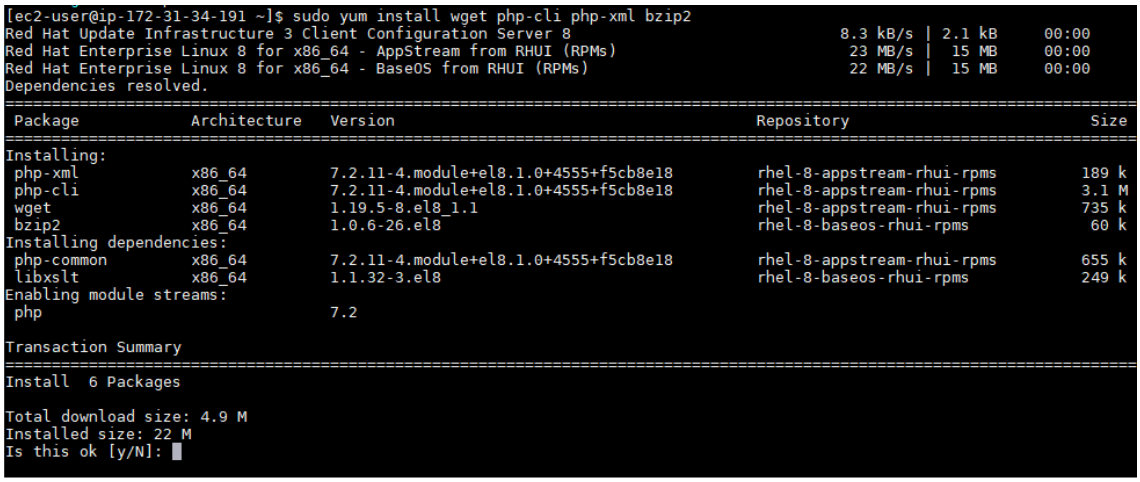


Figura 35: Instal·lacions prèvies a Red Hat Enterprise Linux.

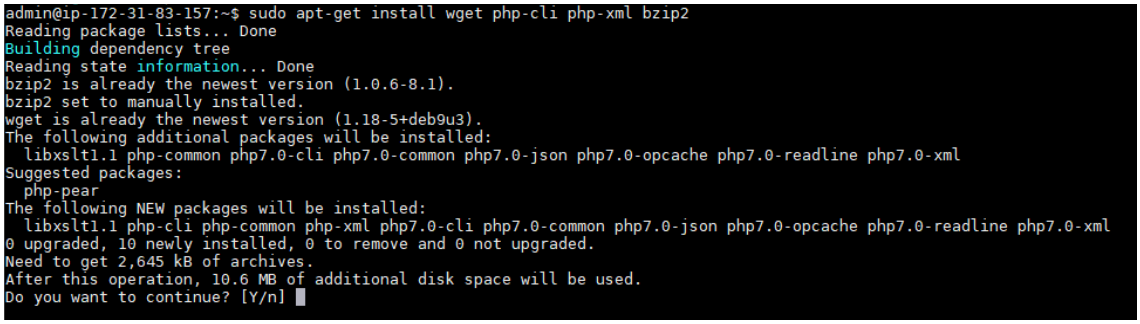


Figura 36: Instal·lacions prèvies a Debian.

```
ubuntu@ip-172-31-85-159:~$ sudo apt-get install wget php-cli php-xml bzip2
Reading package lists... Done
Building dependency tree
Reading state information... Done
bzip2 is already the newest version (1.0.6-8.1ubuntu0.2).
bzip2 set to manually installed.
wget is already the newest version (1.19.4-1ubuntu2.2).
wget set to manually installed.
The following additional packages will be installed:
  libsodium23 php-common php7.2-cli php7.2-common php7.2-json php7.2-opcache php7.2-readline php7.2-xml
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libsodium23 php-cli php-common php-xml php7.2-cli php7.2-common php7.2-json php7.2-opcache php7.2-readline php7.2-xml
0 upgraded, 10 newly installed, 0 to remove and 18 not upgraded.
Need to get 2760 kB of archives.
After this operation, 13.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Figura 37: Instal·lacions prèvies a Ubuntu.

Tot seguit per a la instal·lació de la suite es procedeix amb les següents comandes:

```
wget https://phoronix-test-suite.com/releases/phoronix-test-suite-9.6.0.tar.gz
tar xvfz phoronix-test-suite-9.6.0.tar.gz
# cd phoronix-test-suite
# ./install-sh
```

```
[ec2-user@ip-172-31-34-191 ~]$ wget https://phoronix-test-suite.com/releases/phoronix-test-suite-9.6.0.tar.gz
--2020-04-23 17:57:00--  https://phoronix-test-suite.com/releases/phoronix-test-suite-9.6.0.tar.gz
Resolving phoronix-test-suite.com (phoronix-test-suite.com)... 162.252.83.89
Connecting to phoronix-test-suite.com (phoronix-test-suite.com)|162.252.83.89|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1737278 (1.7M) [application/x-gzip]
Saving to: 'phoronix-test-suite-9.6.0.tar.gz'

phoronix-test-suite-9.6.0.tar. 100%[===================================================>]   1.66M  7.11MB/s    in 0.2s

2020-04-23 17:57:01 (7.11 MB/s) - 'phoronix-test-suite-9.6.0.tar.gz' saved [1737278/1737278]

[ec2-user@ip-172-31-34-191 ~]$
```

Figura 38: Instal·lació Phoronix Test Suite a Red Hat Enterprise Linux.

```
admin@ip-172-31-83-157:~$ wget https://phoronix-test-suite.com/releases/phoronix-test-suite-9.6.0.tar.gz
--2020-04-23 17:57:03--  https://phoronix-test-suite.com/releases/phoronix-test-suite-9.6.0.tar.gz
Resolving phoronix-test-suite.com (phoronix-test-suite.com)... 162.252.83.89
Connecting to phoronix-test-suite.com (phoronix-test-suite.com)|162.252.83.89|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1737278 (1.7M) [application/x-gzip]
Saving to: 'phoronix-test-suite-9.6.0.tar.gz'

phoronix-test-suite-9.6.0.tar. 100%[===================================================>]   1.66M  7.88MB/s    in 0.2s

2020-04-23 17:57:04 (7.88 MB/s) - 'phoronix-test-suite-9.6.0.tar.gz' saved [1737278/1737278]

admin@ip-172-31-83-157:~$
```

Figura 39: Instal·lació Phoronix Test Suite a Debian.

```
[ec2-user@ip-172-31-34-191 phoronix-test-suite]$ sudo ./install-sh
which: no xdg-mime in (/sbin:/bin:/usr/sbin:/usr/bin)

Phoronix Test Suite Installation Completed

Executable File: /usr/bin/phoronix-test-suite
Documentation: /usr/share/doc/phoronix-test-suite/
Phoronix Test Suite Files: /usr/share/phoronix-test-suite/
```

Figura 40: Execució del script d'instal·lació Phoronix Test Suite a Red Hat Enterprise Linux.

```
admin@ip-172-31-83-157:~/phoronix-test-suite$ sudo ./install-sh
-e
Phoronix Test Suite Installation Completed

Executable File: /usr/bin/phoronix-test-suite
Documentation: /usr/share/doc/phoronix-test-suite/
Phoronix Test Suite Files: /usr/share/phoronix-test-suite/
```

Figura 41: Execució del script d'instal·lació Phoronix Test Suite a Debian.

```
ubuntu@ip-172-31-85-159:~/phoronix-test-suite$ sudo ./install-sh
-e
Phoronix Test Suite Installation Completed

Executable File: /usr/bin/phoronix-test-suite
Documentation: /usr/share/doc/phoronix-test-suite/
Phoronix Test Suite Files: /usr/share/phoronix-test-suite/

ubuntu@ip-172-31-85-159:~/phoronix-test-suite$
```

Figura 42: Execució del script d'instal·lació Phoronix Test Suite a Ubuntu.

En aquest punt es verifica que es disposa de la suite de proves en els sistemes:

```
admin@ip-172-31-83-157:~$ /usr/bin/phoronix-test-suite list-available-suites


Updated OpenBenchmarking.org Repository Index
pts: 394 Distinct Tests, 1472 Test Versions, 41 Suites
Updated Test Available:   pts/aircrack-ng        v1.2.1
New Test Available:       pts/blosc              v1.0.0
Updated Test Available:   pts/build-linux-kernel v1.10.1
Updated Test Available:   pts/fio                v1.13.2
Updated Test Available:   pts/lczero             v1.4.0
Updated Test Available:   pts/svt-av1            v2.2.1
Updated Test Available:   pts/svt-vp9            v1.2.2
Updated Suite Available:  pts/compression        v1.1.1
Updated OpenBenchmarking.org Repository Index
system: 32 Distinct Tests, 85 Test Versions, 0 Suites

Updated OpenBenchmarking.org Repository Index
git: 8 Distinct Tests, 9 Test Versions, 0 Suites


Phoronix Test Suite v9.6.0
Available Suites

  pts/audio-encoding              - Audio Encoding                  System
  pts/av1                         - AV1                             System
  pts/bioinformatics              - Bioinformatics                  System
  pts/chess                       - Chess Test Suite                Processor
  pts/compilation                 - Timed Code Compilation          System
  pts/compiler                    - C/C++ Compiler Tests            Processor
  pts/compression                 - Compression Tests               Processor
  pts/creator                     - Creator Workloads               System
  pts/cryptography                - Cryptography                    Processor
  pts/database                    - Database Test Suite             System
  pts/desktop-graphics            - Desktop Graphics                System
  pts/disk                        - Disk Test Suite                 Disk
  pts/gaming                      - Gaming                          System
  pts/gaming-closed               - Closed-Source Gaming            System
  pts/gaming-free                 - Free Software Gaming            System
  pts/hpc                         - HPC - High Performance Computing System
  pts/imaging                     - Imaging                         System
  pts/java                        - Java                            System
  pts/kernel                      - Common Kernel Benchmarks        System
  pts/linear-algebra              - Linear Algebra                  System
  pts/machine-learning            - Machine Learning                System
  pts/memory                      - Memory Test Suite               Memory
  pts/molecular-dynamics          - Molecular Dynamics              System
  pts/mpi                         - MPI Benchmarks                  System
  pts/multicore                   - Multi-Core                      Processor
  pts/network                     - Networking Test Suite           Network
  pts/oneapi                      - Intel oneAPI                    System
  pts/opencl                      - OpenCL                          System
  pts/python                      - Python                          System
  pts/raytracing                  - Raytracing                      System
  pts/renderers                   - Renderers                       Processor
  pts/scientific-computing        - Scientific Computing            System
  pts/server                      - Server                          System
* pts/single-threaded              - Single-Threaded                 System
  pts/unigine                     - Unigine Test Suite              Graphics
  pts/video-encoding              - Video Encoding                  System
  pts/workstation                 - Common Workstation Benchmarks   System

* Indicates a partially supported suite.


admin@ip-172-31-83-157:~$
```

Figura 43: Comprovació correcta instal·lació Phoronix Test Suite.

5. Selecció de les proves a executar

Per avaluar el rendiment dels sistemes amb SELinux i AppArmor s'han seleccionat una serie de proves de la suite Phoronix Test Suite ja que la suite disposa d'un gran nombre de tests i conjunts o suites. La principal diferència és que en una suite de les disponibles, hi ha una agrupació d'una serie de tests que avaluen característiques similars, p.e. la suite server inclou tests de tecnologies i funcionalitats típiques d'un servidor.

Les proves de rendiment s'han executat amb els següents tests disponibles de la suite:

- pts/apache
- pts/fftw
- pts/nginx
- pts/openssl
- pts/encode-mp3
- pts/ramspeed-1.4.2
- pts/postmark
- pts/ipc-benchmark
- pts/ctx-clock
- pts/sysbench

A continuació es detallen en que consisteix cadascun d'aquests test:

5.1. pts/apache

Aquesta prova mesura quantes sol·licituds pot suportar un sistema determinat quan es realitzen 1.000.000 sol·licituds amb 100 sol·licituds simultàniament.

5.2. pts/fftw

FTW és una biblioteca de subrutina C per calcular la transformada de discreta de Fourier en una o més dimensions.

5.3. pts/nginx

Aquesta prova mesura quantes sol·licituds pot suportar un sistema determinat quan es realitzen 2.000.000 sol·licituds amb 500 sol·licituds simultàniament.

5.4. pts/openssl

Aquest test mesura el rendiment de 4096 bits RSA d'OpenSSL.

5.5. pts/encode-mp3

Aquesta prova mesura el temps necessari per codificar un fitxer WAV en format MP3.

5.6. pts/ramspeed-1.4.2

Conjunt de proves destinades a provar el rendiment de la memòria del sistema (RAM).

5.7. pts/postmark

Prova per simular proves de petits fitxers similars a les tasques que suporten els servidors web i de correu. Aquest perfil realitza 25.000 transaccions amb 500 fitxers simultàniament amb les mides dels fitxers entre 5 i 512 quilobytes.

5.8. pts/ipc-benchmark

Conjunt de proves per a la comunicació entre processos Linux.

5.9. pts/ctx-clock

Ctx_clock és un programa de prova senzill per mesurar el temps de canvi de context en cicles de rellotge.

5.10. pts/sysbench

Es tracta d'un punt de referència de Sysbench amb subproves de CPU i memòria.

6. Execució de les proves i resultats

En aquest apartat es presenten els resultats de la execució dels jocs de proves i la interpretació que es pot extreure d'ells.

Per cada joc de proves, es mostraran els resultats agrupats en funció del número de processadors del sistema on s'han executat les proves. En aquest sentit, es començara amb les proves executades en sistemes amb un sol nucli en el processador que respon a la següent configuració:

Processor	Intel Xeon E5-2676 v3 (1 Core)
Motherboard	Xen HVM domU (4.2.amazon BIOS)
Chipset	Intel 440FX 82441FX PMC

Després, es presenten els jocs de proves dels sistemes amb 4 nuclis d'execució en el processador amb la següent configuració:

Processor	Intel Xeon E5-2686 v4 (4 Cores)
Motherboard	Xen HVM domU (4.2.amazon BIOS)
Chipset	Intel 440FX 82441FX PMC

Adjunt a cada resultat, hi ha una taula amb les característiques especifiques referents a la resta de paràmetres que configuren el sistema i si el sistema te o no te els mòduls de seguretat, SELinux o AppArmor activats o desactivats.

6.1. pts/apache

Entorn monocore

	rh329042020	rh330042020	deb0117052020	dbn0117052020	ubu0103052020	UBU0104052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

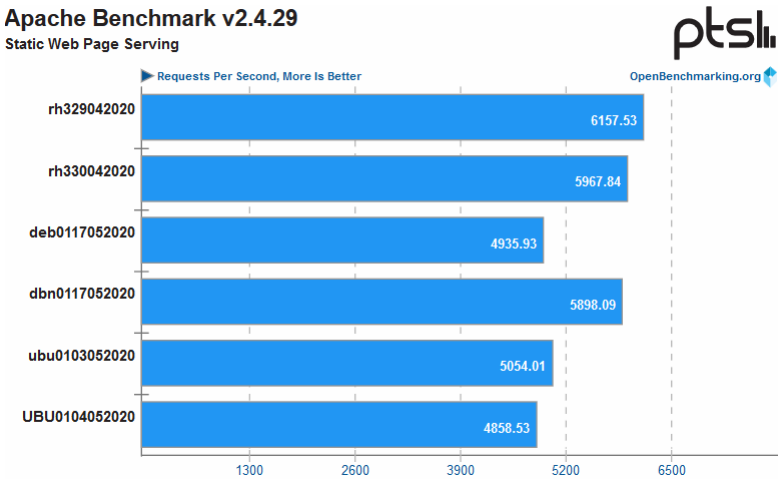


Figura 44: Resultats proves Apache entorn monocore.

En aquest primer joc de proves s'observa, que en el sistema amb Debian el rendiment ha sigut millor quan SELinux estava desactivat. Pel contrari, en el cas de Red Hat i Ubuntu es dona millor rendiment quan els sistemes tenen activats els mòduls de seguretat.

Entorn Multicore

	rhe0109052020	re0109052020	deb0109052020	db0109052020	ubu0109052020	ub0109052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

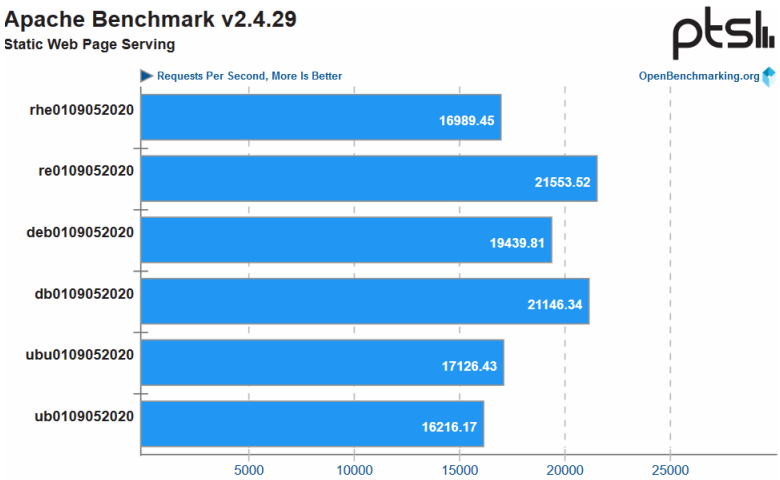


Figura 45: Resultats proves Apache entorn multicore.

En aquesta prova observem en els resultats, que per als sistemes amb SELlinux el rendiment ha sigut millor quan aquest ha estat desactivat. Per contra, en el sistema Ubuntu, el rendiment ha sigut millor amb AppArmor activat.

6.2. pts/fftw

Entorn monocore

	rh429042020	rh430042020	deb0217052020	dbn0217052020	ubu0203052020	ubu0204052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

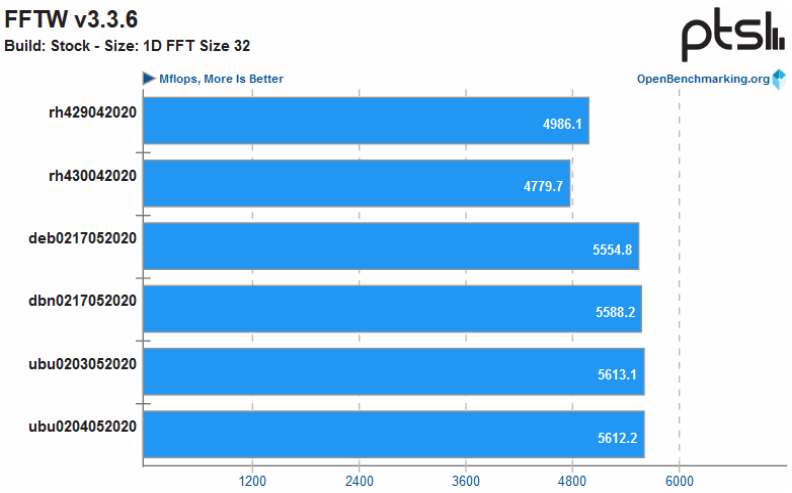


Figura 46: Resultats proves FFTW entorn monocore.

Per els sistemes Debian i Ubuntu, pràcticament no existeix diferències en el rendiment respecte a aquest joc de proves. Per contrari, en Red Hat s'observa millor rendiment amb SELinux activat.

Entorn multicore

	rhe0209052020	rh0209052020	deb0209052020	db0209052020	ubu0209052020	ub0209052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

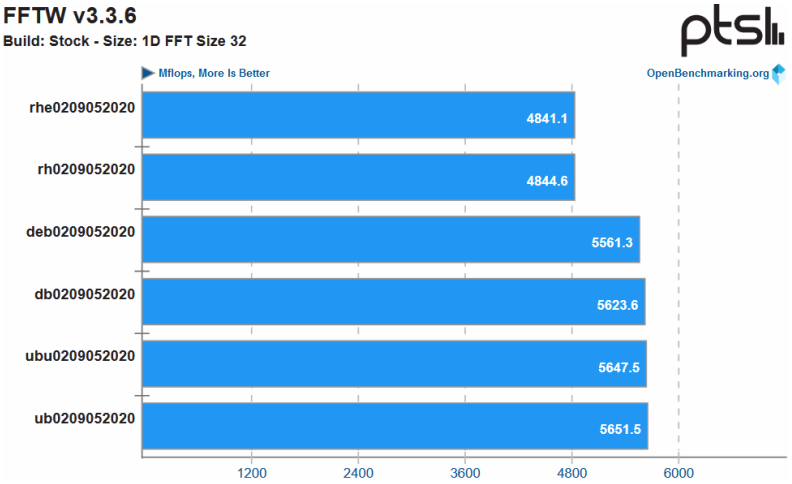


Figura 47: Resultats proves FFTW entorn multicore.

Observem que en general els resultats han sigut molt semblants, per tant, podem concloure que per a aquest joc de proves el fet d'executar-se amb els mòduls de seguretat activats, no influeixen en el seu rendiment.

6.3. pts/nginx

Entorn monocore

	rh529042020	rh530042020	deb0317052020	dbn0317052020	ubu0303052020	ubu0304052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

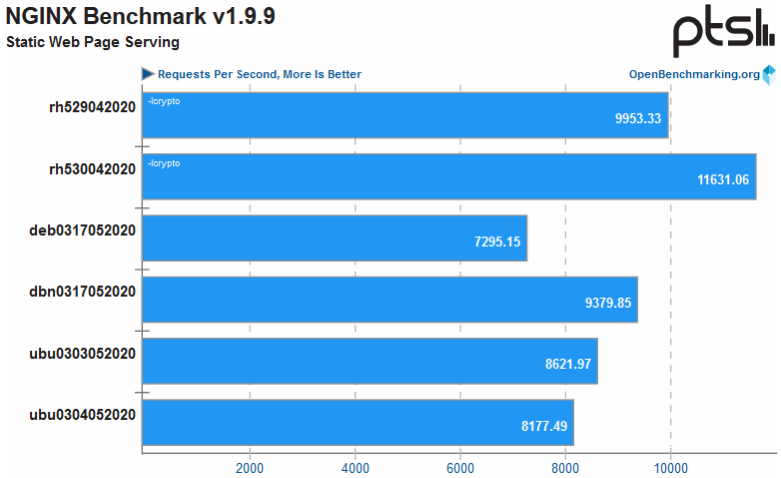


Figura 48: Resultats proves NGINX entorn monocore.

En els sistemes amb SELinux el rendiment és millor quan el mòdul està desactivat. Per contra, a Ubuntu amb AppArmor activat, el resultat és millor.

Entorn multicore

	rhe0309052020	rh0309052020	deb0309052020	db0309052020	ubu0309052020	ub0309052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

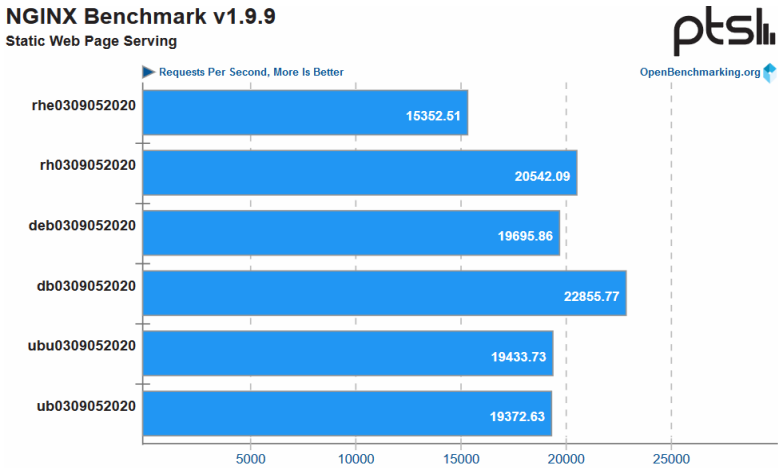


Figura 49: Resultats proves NGINX entorn multicore.

S'observa que en els sistemes amb SELinux activat, el rendiment s'ha vist penalitzat d'una forma significativa. Per contra, l'impacte en el rendiment de AppArmor en els sistemes Ubuntu és inapreciable. Per aquest joc de proves, AppArmor obté millors resultats que SELinux.

6.4. pts/openssl

Entorn monocore

	rh629042020	rh630042020	deb0417052020	dbn0417052020	ubu0403052020	ubu0404052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

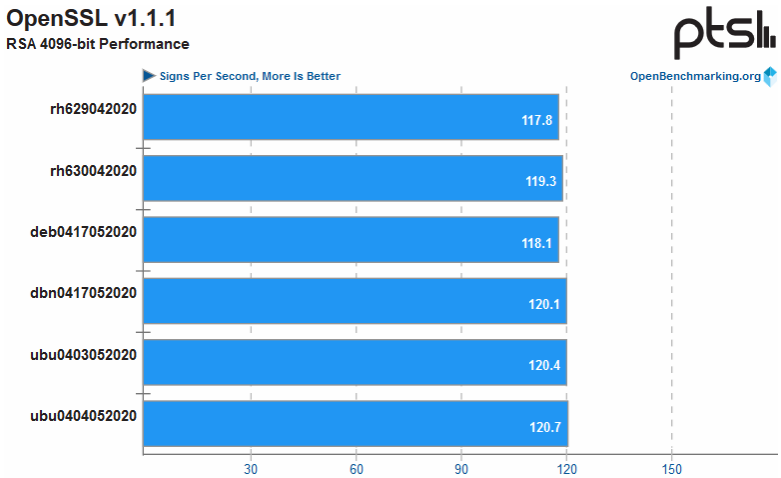


Figura 50: Resultats proves OpenSSL entorn monocore.

En aquest joc de proves, les diferències observades per a cada escenari són molt petites. Es pot concloure que el fet que els mòduls de seguretat estiguin activats o desactivats, no suposa un impacte, en aquest cas.

Entorn multicore

	rhe0409052020	rh0409052020	deb0409052020	db0409052020	ubu0409052020	ub0409052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

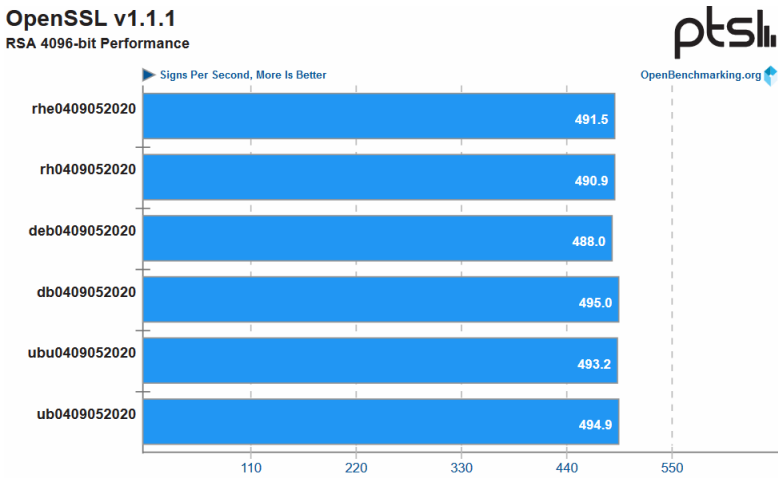


Figura 51: Resultats proves OpenSSL entorn multicore.

Per aquest joc de proves, no s'observa impacte en la execució pel fet de que SELinux o AppArmor estiguin activats en els sistemes.

6.5. pts/encode-mp3

Entorn monocore

	rh429042020	rh430042020	deb0517052020	dbn0517052020	ubu0503052020	ubu0504052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

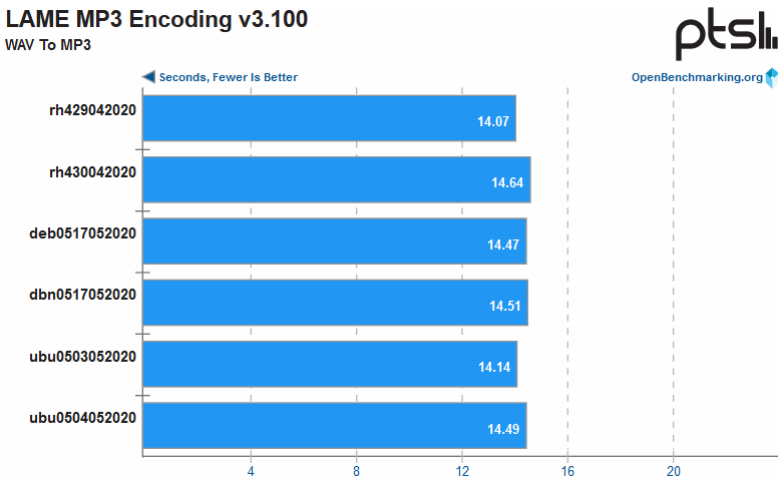


Figura 52: Resultats proves Encode/mp3 entorn monocore.

Per aquest joc de proves, els resultats són lleugerament millors quan SELinux i AppArmor estan activats.

Entorn multicore

	rhe0509052020	rh0509052020	deb0509052020	db0509052020	ubu0509052020	ub0509052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

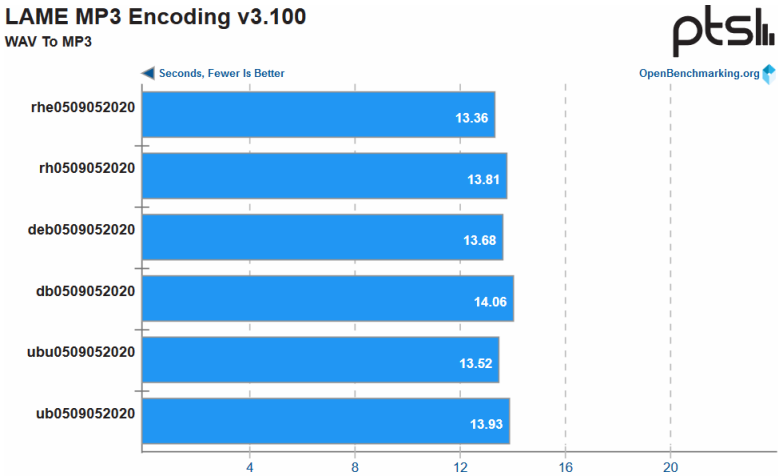


Figura 53: Resultats proves Encode/mp3 entorn multicore.

En aquest cas, s'observa que les proves realitzades en els sistemes amb els mòduls de seguretat activats, han donat millors resultats de rendiment que quan aquests mòduls han estat desactivats. Tot i així, la diferència és molt poca.

6.6. pts/rampspeed-1.4.2

Entorn monocore

	rh829042020	rh830042020	deb0617052020	dbn0617052020	ubu0603052020	ubu0504052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

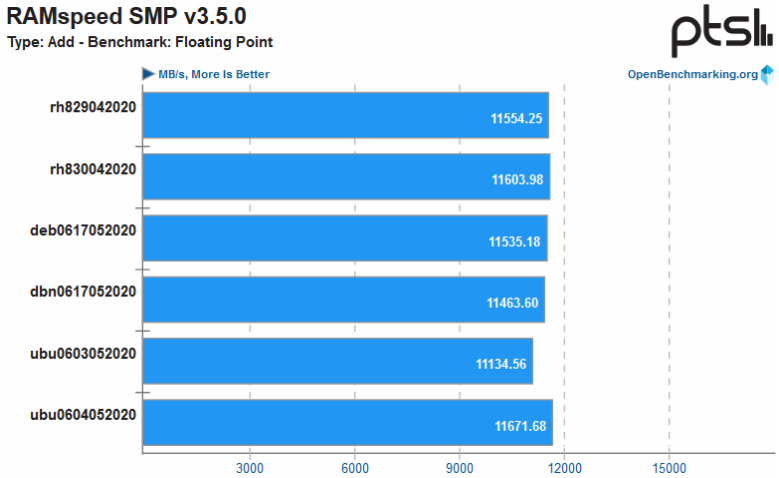


Figura 54: Resultats proves rampspeed-add entorn monocore.

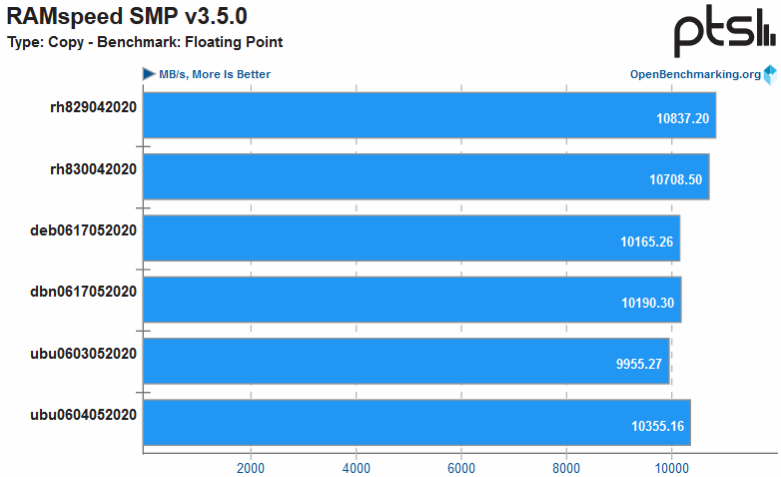


Figura 55: Resultats proves rampspeed-copy entorn monocore.

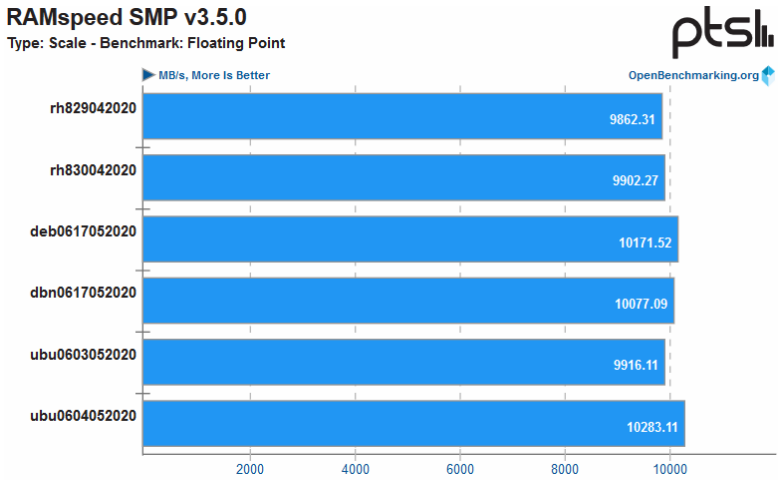


Figura 56: Resultats proves rampspeed-scale entorn monocore.

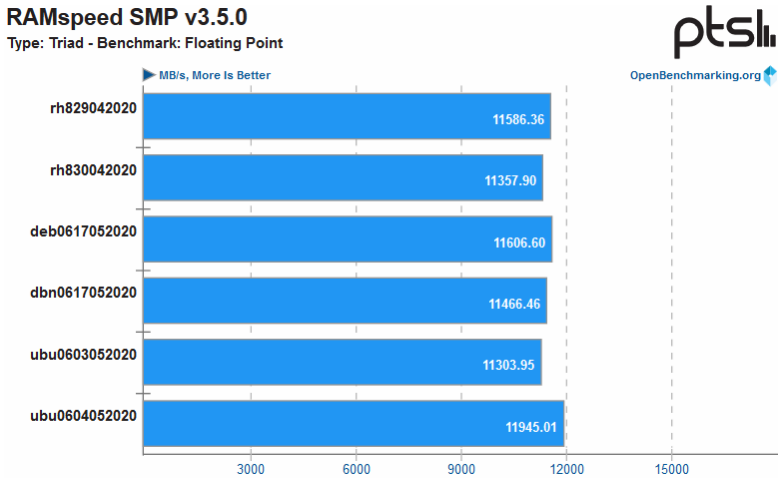


Figura 57: Resultats proves rampspeed-triad entorn monocore.

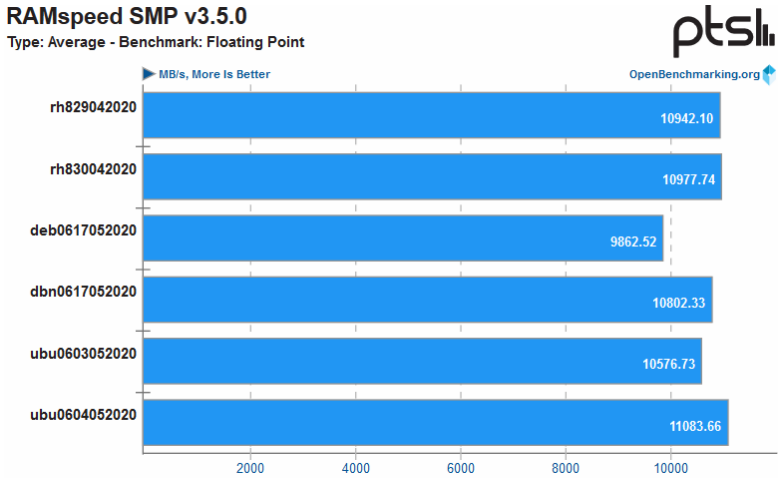


Figura 58: Resultats proves rampspeed-average entorn monocore.

Per cada prova que conforma aquest joc de proves s'observen resultats diferents. En el primer cas, a excecpció del cas de Debian, en els altres dos sistemes els resultats són millors amb SELinux i AppArmor desactivats. En la segona prova, és el sistema Red Hat el que divergeix del comportament dels altres dos, obtenint millor resultat amb SELinux activat. Per la tercera prova, torna a ser el

sistema Debian el que difereix dels altres dos. En el cas de la quarta, és el sistema amb Ubuntu amb AppArmor desactivat qui obté millor resultats. I finalment, per a la cinquena prova, si que s'observa el mateix patro per als tres sistemes. En aquest cas, s'obté millor rendiment amb els mòduls desactivats.

Entorn multicore

	rhe0609052020	rh0609052020	deb0609052020	db0609052020	ubu0609052020	ub0609052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

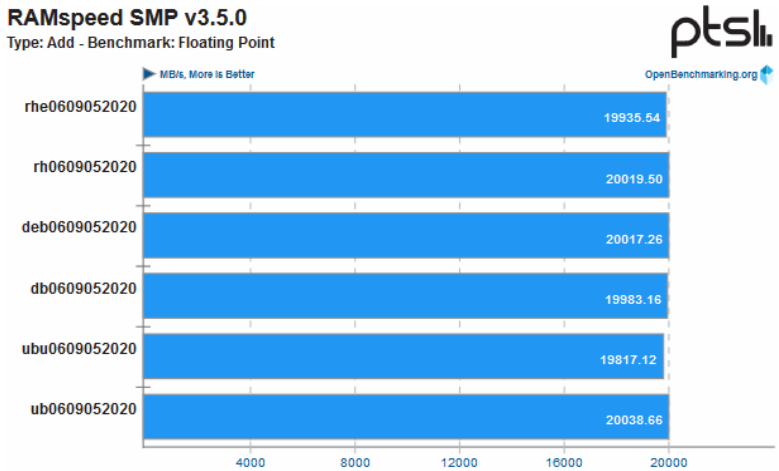


Figura 59: Resultats proves rampspeed-add entorn multicore.

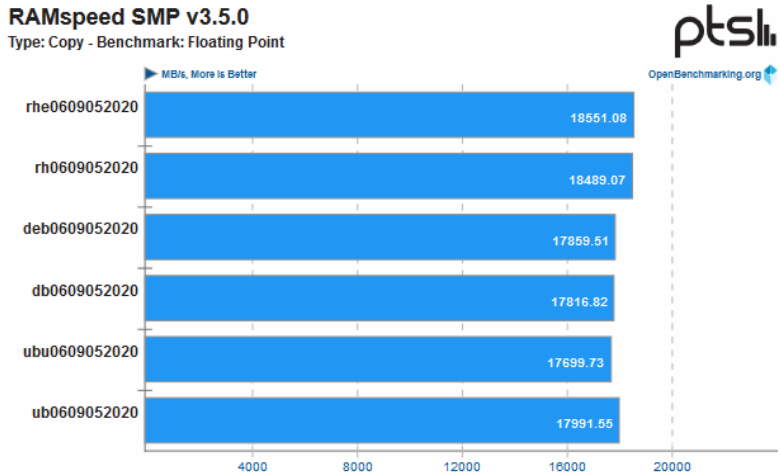


Figura 60: Resultats proves rampspeed-copy entorn multicore.

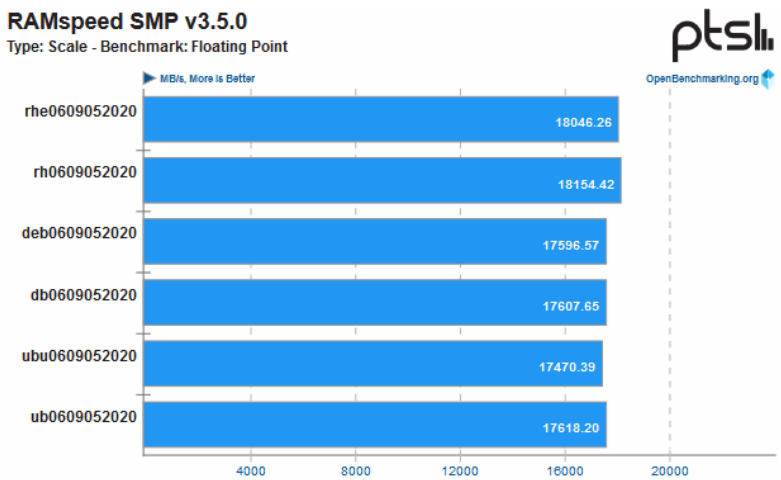


Figura 61: Resultats proves rampspeed-scale entorn multicore.

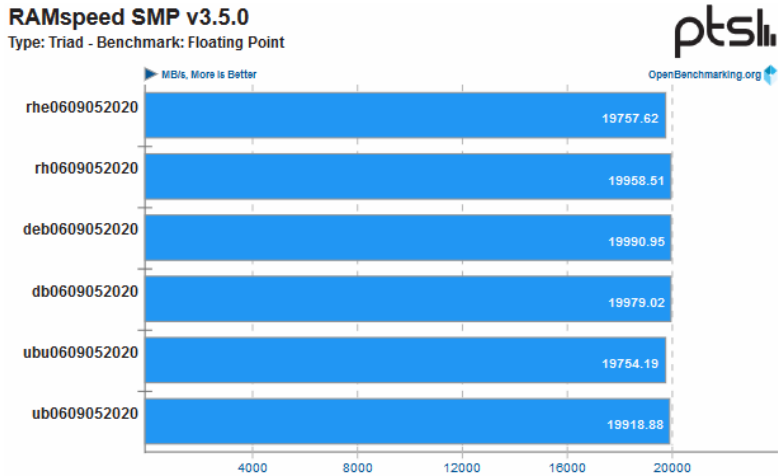


Figura 62: Resultats proves rampspeed-triad entorn multicore.

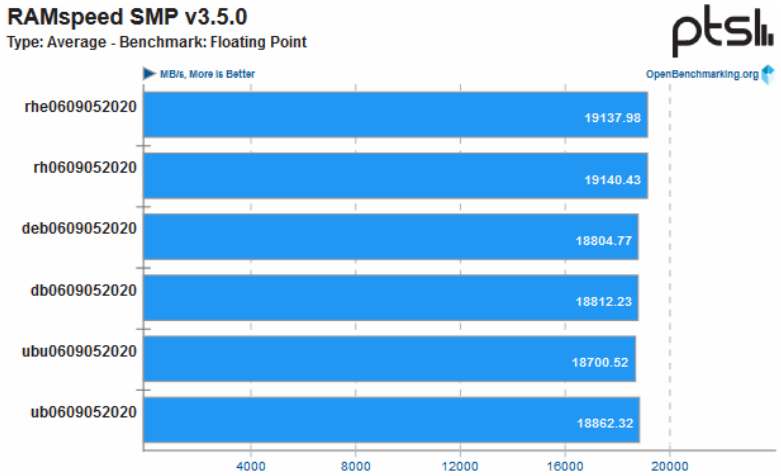


Figura 63: Resultats proves rampspeed-average entorn multicore.

En les cinc proves que componen el joc actual, s'observa que en tots els casos, el rendiment és lleugerament millor en els sistemes amb els mòduls de seguretat desactivats. També podem observar que en tots els casos, el sistema que pitjor resultat obté és l'Ubuntu amb AppArmor activat.

6.7. pts/postmark

Entorn monocore

	rh929042020	rh930042020	deb0717052020	dbn0717052020	ubu0703052020	ubu0704052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

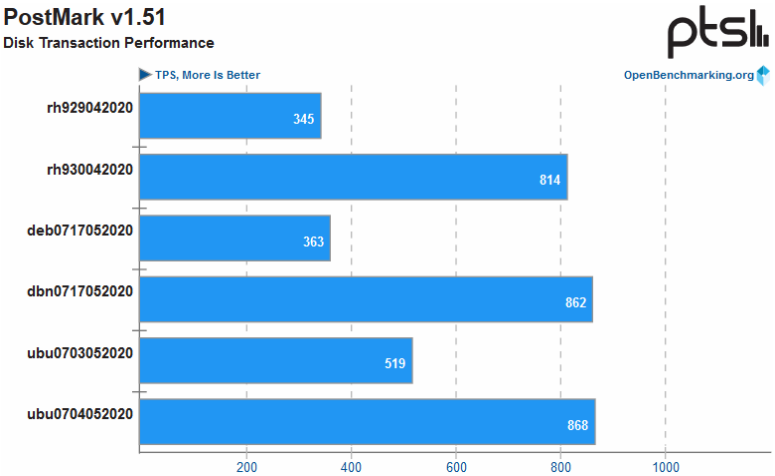


Figura 64: Resultats proves postmark entorn monocore.

En aquesta ocasió per tots els sistemes, el rendiment és substancialment millor quan els mòduls de seguretat es troben desactivats.

Entorn multicore

	rhe0709052020	rh0709052020	deb0709052020	db0709052020	ubu0709052020	ub0709052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

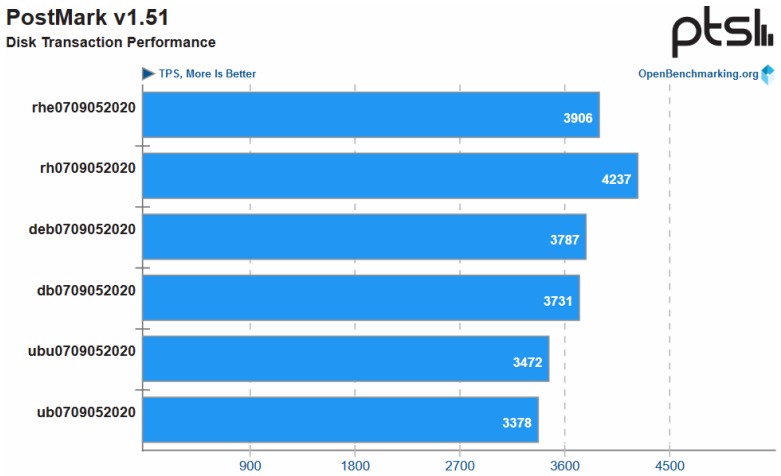


Figura 65: Resultats proves postmark entorn multicore.

En aquest joc de proves, s'observa que el sistema Red Hat Enterprise Linux amb SELinux desactivat és el que millor resultat obté. En canvi en els sistemes Debian i Ubuntu, els resultats són lleugerament millors quan SELinux i AppArmor estan activats.

6.8. pts/ipc-benchmark

Entorn monocore

	rh929042020	rh930042020	deb0717052020	dbn0717052020	ubu0703052020	ubu0704052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

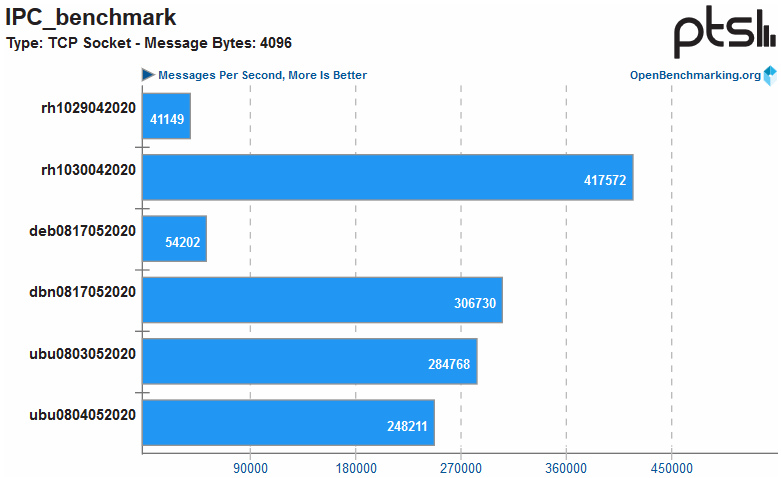


Figura 66: Resultats proves ipc_benchmark tcp entorn monocore.

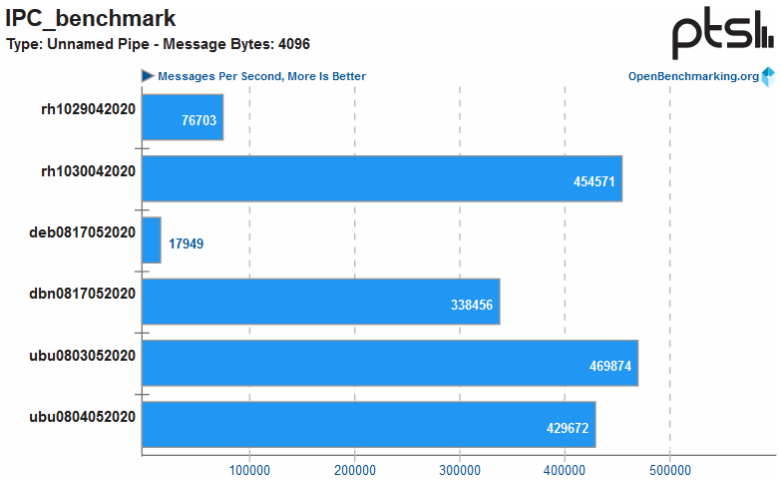


Figura 67: Resultats proves ipc_benchmark unamed pipe entorn monocore.

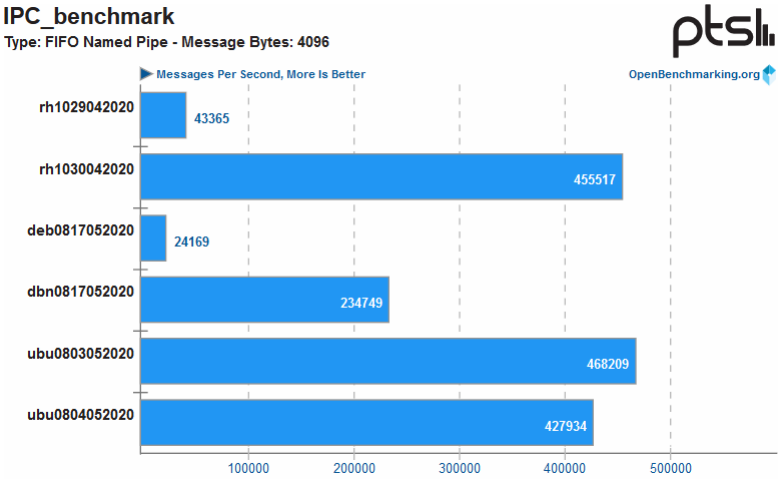


Figura 68: Resultats proves ipc_benchmark named pipe entorn monocore.

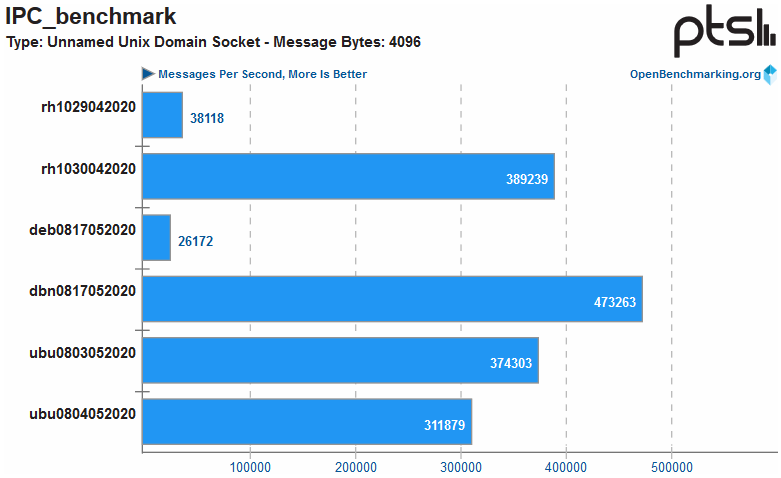


Figura 69: Resultats proves ipc_benchmark unnamed unix entorn monocore.

En els sistemes amb SELinux, s'observa que aquest joc de proves quan aquest mòdul està activat penalitza en gran mesura el rendiment del sistema en la seva execució. Per contra, en els sistemes Ubuntu, els resultats obtinguts indiquen que el rendiment és millor quan AppArmor està excutant-se.

Entorn multicore

	rhe0809052020	rh0809052020	deb0809052020	db0809052020	ubu0809052020	ub0809052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

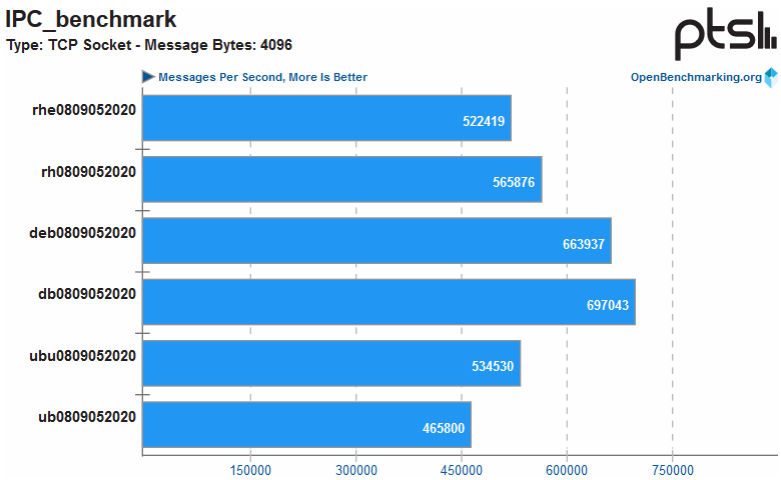


Figura 70: Resultats proves ipc_benchmark tcp entorn multicore.

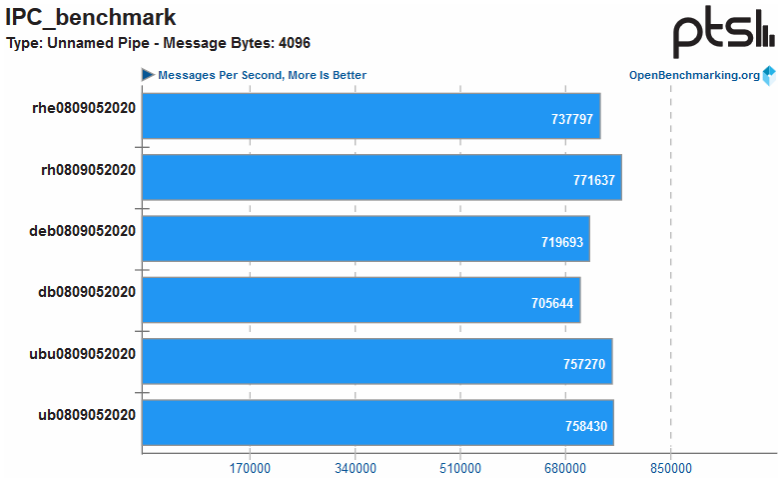


Figura 71: Resultats proves ipc_benchmark unnamed pipe entorn multicore.

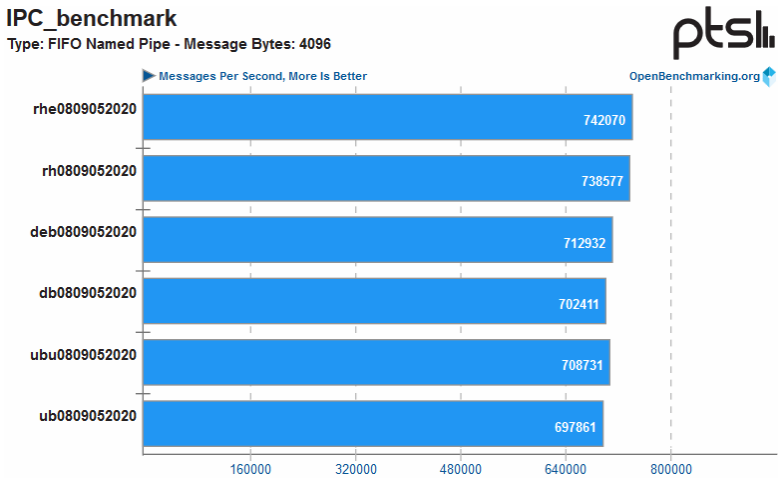


Figura 72: Resultats proves ipc_benchmark named pipe entorn multicore.

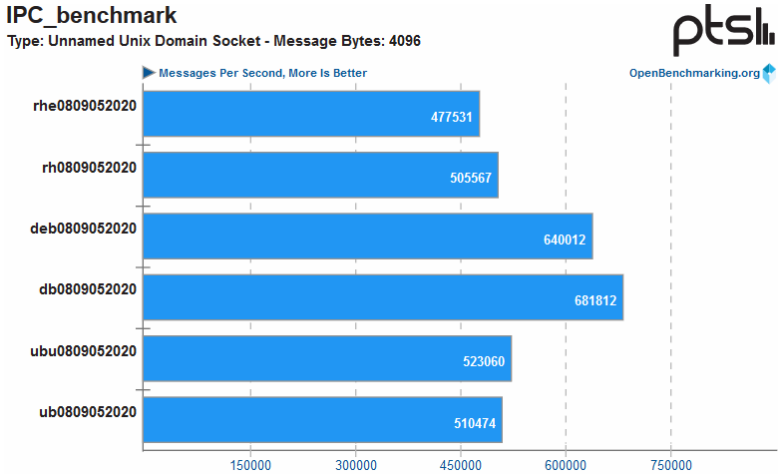


Figura 73: Resultats proves ipc_benchmark unnamed unix entorn multicore.

Els resultats d'aquest joc de proves són força diferenciats respecte a les quatre proves específiques que s'han executat. En la primera d'elles s'observa que el sistema Ubuntu amb AppArmor desactivat és el que pitjor resultat ha obtingut, a diferència dels sistemes on SELinux estava desactivat que han tingut millors resultats.

Per la segona prova, és el sistema Debian amb SELinux desactivat qui obté els pitjors resultats, al contrari que el sistema RHEL i Ubuntu amb els mòduls desactivats que obtenen millor rendiment.

En la tercera prova, si que s'aprecia un comportament uniforme en els resultats, sent millors en els casos que SELinux o AppArmor estaven activats. I finalment, a la quarta prova, s'observa que en els sistemes amb SELinux desactivat els resultats són millors però en canvi amb Ubuntu el rendiment és millor si està AppArmor activat.

6.9. pts/ctx-clock

Entorn monocore

	rh1129042020	rh1130042020	deb0917052020	dbn0917052020	ubu0903052020	ubu0904052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB

Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

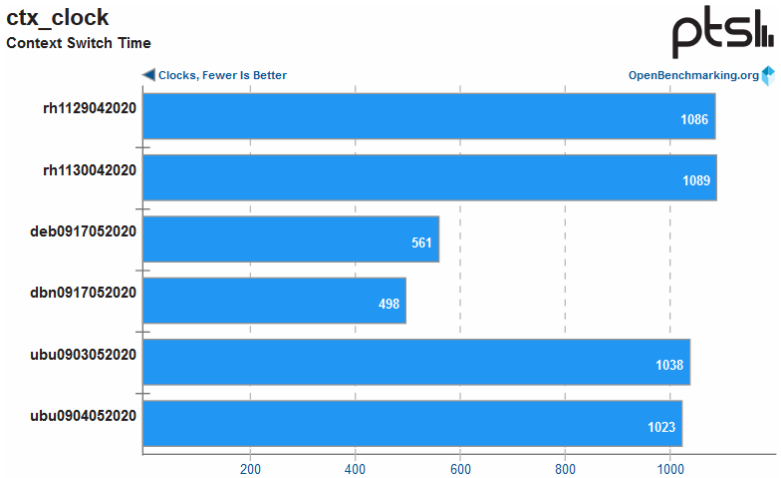


Figura 74: Resultats proves ctx_clock entorn monocore.

Els resultats mostren millor resultat a Debian i Ubuntu quan SELinux i AppArmor estan desactivats. Per contra a Red Hat, la diferència és bastant petita sent millor quan està activat.

Entorn multicore

	rhe0909052020	rh0909052020	deb0909052020	db0909052020	ubu0909052020	ub0909052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

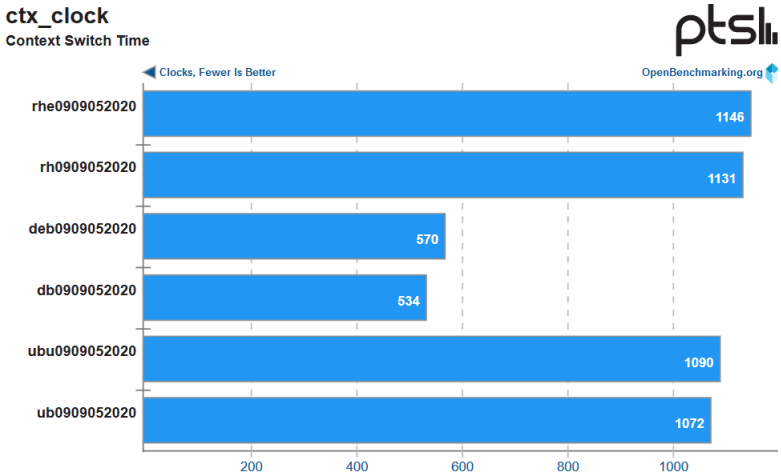


Figura 75: Resultats proves ctx_clock entorn multicore.

En tots els sistemes el rendiment és lleugerament millor quan els mòduls de seguretat estan desactivats. S'observa que per als sistemes amb Debian com a sistema operatiu, aquesta prova dona un resultats molt millor que per a la resta de sistemes operatius.

6.10. pts/sysbench

Entorn monocore

	rh1329042020	compare1	deb1117052020	dbn1117052020	ubu1103052020	ubu1104052020
Memory	814MB	814MB	1024MB	1024MB	1024MB	1024MB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Disabled	SELinux Enabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

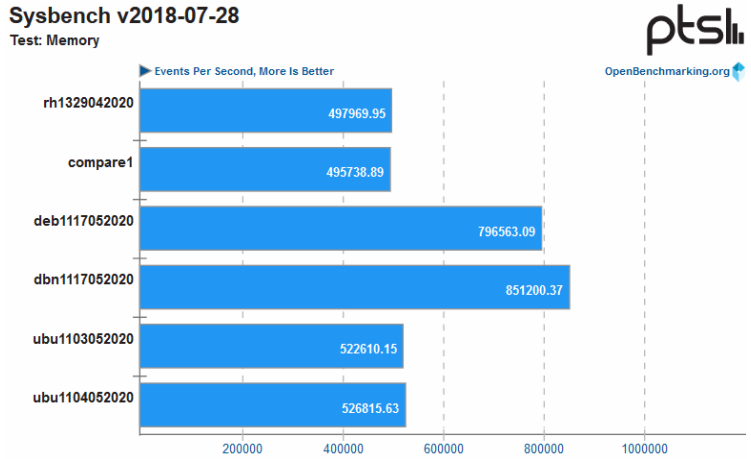


Figura 76: Resultats proves sysbench memory entorn monocore.

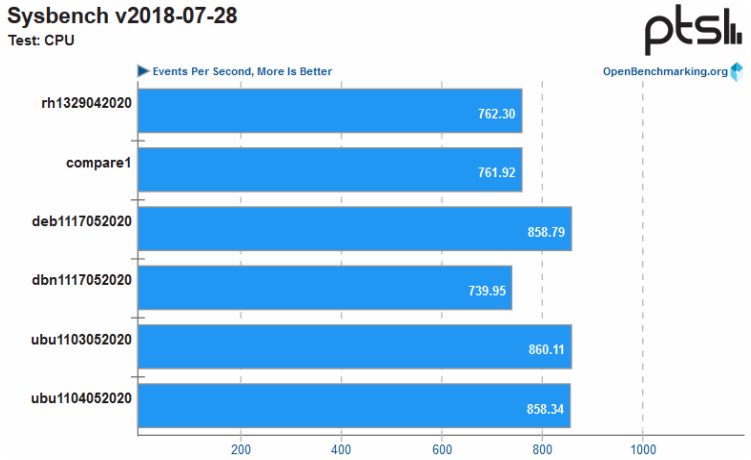


Figura 77: Resultats proves sysbench cpu entorn monocore.

Per a aquest joc de proves, els resultats en els casos dels sistemes amb Red Hat i Ubuntu són pràcticament iguals, es pot concloure que no hi ha impacte dels mòduls de seguretat. Per contrari, en

el sistema Debian, s'observa millor rendiment quan SELinux es troba activat a la prova de disc, en canvi el resultat és millor estant desactivat a la prova de memòria.

Entorn multicore

	rhe1109052020	rh1109052020	deb1109052020	db1109052020	ubu1109052020	ub1109052020
Memory	16GB	16GB	16GB	16GB	16GB	16GB
Disk	10GB	10GB	8GB	8GB	8GB	8GB
OS	Red Hat Enterprise Linux 8.2	Red Hat Enterprise Linux 8.2	Debian 9.12	Debian 9.12	Ubuntu 18.04	Ubuntu 18.04
Kernel	4.18.0-193.el8.x86_64 (x86_64)	4.18.0-193.el8.x86_64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.9.0-8-amd64 (x86_64)	4.15.0-1065-aws (x86_64)	5.3.0-1017-aws (x86_64)
Compiler	GCC 8.3.1 20191121	GCC 8.3.1 20191121	GCC 6.3.0 20170516	GCC 6.3.0 20170516	GCC 7.5.0	GCC 7.5.0
File-System	xfs	xfs	ext4	ext4	ext4	ext4
System Layer	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon	Xen HVM domU 4.2.amazon
Security Module	SELinux Enabled	SELinux Enabled	SELinux Disabled	SELinux Disabled	AppArmor Enabled	AppArmor Disabled

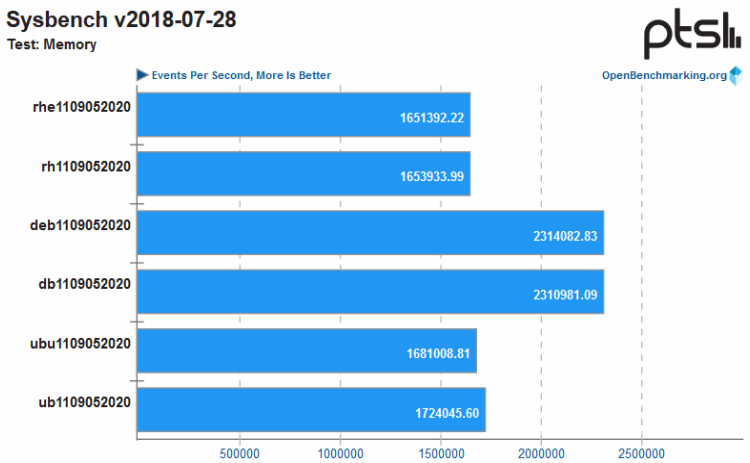


Figura 78: Resultats proves sysbench memory entorn multicore.

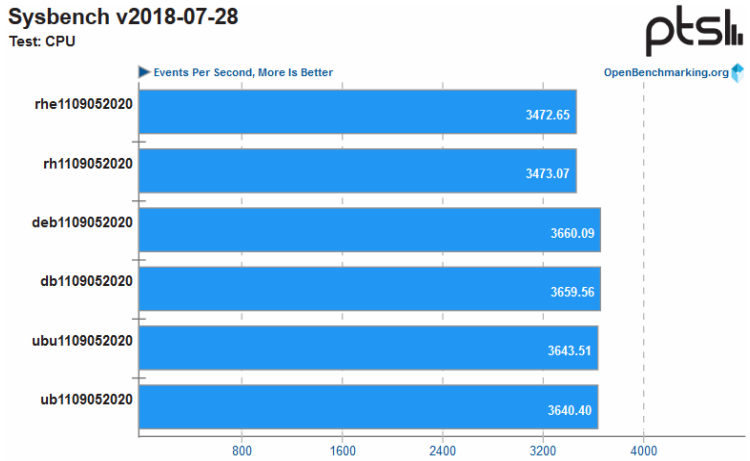


Figura 79: Resultats proves sysbench cpu entorn multicore.

Observant els resultats d'aquest joc de proves, les diferències en quant al rendiment es poden considerar no significatives.

7. Conclusions

Un cop presentats tots els resultats, podem diferenciar tres grups. El primer grup correspon a aquells on els resultats són pràcticament iguals en els sistemes monocore i multicore. Es a dir, els resultats descriuen que el rendiment estant activats o desactivats els mòduls de seguretat en el sistema són pràcticament iguals, independentment del nombre de cores de CPU i la memòria RAM del sistema.

Els jocs de proves que segueixen aquest patró són: fftw, openssl i ctx_clock. En aquest mateix grup es pot incloure el joc sysbench, ja que hi ha una lleugera discrepància en el resultat del test de CPU per al sistema Debian de l'entorn monocore.

Tot seguit, hi ha el grup on el fet que els mòduls de seguretat estiguin activats o desactivats influeix molt clarament en el rendiment del sistema. Aquest seria el cas del jocs encode-mp3, postmark i ipc-benchmark, on els resultats observats són molt millors quan els respectius mòduls de seguretat estan desactivats en els sistemes a l'entorn monocore.

Aquestes diferències són menys significatives en l'entorn multicore. Això és molt destacable per als test postmark i ipc-benchmark. En el cas de encode-mp3, les diferències no són tan marcades com en els dos casos anteriors.

Pel joc de proves ramspeed, atesa la seva diversitat de proves s'observa que en general els resultats són lleugerament millors quan els mòduls estan desactivats. Però la diferència és poca per tant, es pot considerar que la discrepància és petita.

Finalment, trobem el grup de proves format per els jocs d'Apache i Nginx. Els resultats descriuen un comportament diferent respecte a la resta de jocs. En el cas dels sistemes Ubuntu, tant per als entorns monocore com multicore, el rendiment és millor quan AppArmor es troba activat.

Per contra, els sistemes Debian obtenen millor resultat en ambdues proves quan SELinux està desactivat. En canvi, els sistemes Red Hat també s'obtenen millors resultats en el dos entorns amb SELinux desactivat, a excepció de l'entorn monocore on l'Apache millora amb SELinux activat, sent un resultat que es podria classificar com fora del patró.

Podem concloure que el rendiment en els sistemes amb mòduls de seguretat actius, es veu perjudicat de manera poc significativa. Els resultats en l'entorn monocore si que mostren en alguns casos, un impacte que podria alt podent considerar-se preocupant respecte el balanç seguretat versus funcionalitat. Per contra, els resultats dels entorns multicore, s'ajusten més a qualsevol sistema

d’escriptori o servidor actuals. En aquests sistemes les diferències de rendiment són poc significatives.

Les excepcions les trobem en els casos de les proves Apache, Nginx. Les diferències en aquests casos si són significatives en el cas de SELinux, no així per AppArmor. Es pot dir, que AppArmor seria millor opció que SELinux en quant a la protecció del sistema i el prejudici en el rendiment que es pugui presentar.

Això pot ser degut al fet que les operacions dels servidors web, són de les més protegides per els mòduls de seguretat. Per tant, la implementació que AppArmor realitza del control d’accés obligatori per els servidors Apache i Nginx és més òptima, per al rendiment del sistema, que no pas la forma en la que SELinux implementa aquesta protecció per a aquestes dues aplicacions.

Dels resultats de l’entorn monocore, podem extreure com a conclusió que aquest tipus de funcionalitats de seguretat dels mòduls del kernel, poden tenir un impacte considerable en sistemes antic o especialitzats, on l’ús d’aquesta arquitectura en el microprocessador sigui la única opció possible.

Per concloure amb aquest anàlisi, podem dir que les proves en general demostren que és recomanable l’ús dels mòduls del kernel en els sistemes alhora d’implementar el mecanisme de control d’accés obligatori. Atès que els resultats mostren que el balanç del compromís entre rendiment i seguretat en els sistemes amb els mòduls de seguretat activats és acceptable.

En general, si els sistemes són extremadament crítics, com ara sistemes en temps real, l’ús de SELinux seria menys recomanable que el de AppArmor. Inclús potser seria millor cercar altres mètodes de protecció dels sistemes per no penalitzar el rendiment, que tot i que globalment no te un gran l’impacte, sí pot suposar crític per a aquests tipus de sistemes.

Finalment, si s’hagués d’escollir entre un dels dos sistemes, SELinux o Apparmor, els resultats indiquen que aquesta última seria la millor opció. A més a més, cal tenir present que de les dues opcions és la més fàcil de gestionar i, no requereix d’un sistema de fitxers propi. Fet que pot influir en el rendiment del sistema, al haver d’accedir a diferents sistemes de fitxers durant les operacions que es duen a terme amb la supervisió implementada per SELinux.

Appendix

Enllaços a Phoronix on es recullen els resultats individuals i comparatius per a cada test executat en els sistemes de AWS amb els mòduls de seguretat activats o desactivats.

Entorn monocore

RHEL - SELINUX Enabled
https://openbenchmarking.org/result/2004295-NI-RH329042079
https://openbenchmarking.org/result/2004298-NI-RH429042007
https://openbenchmarking.org/result/2004293-NI-RH529042025
https://openbenchmarking.org/result/2004293-NI-RH629042047
https://openbenchmarking.org/result/2004294-NI-RH729042040
https://openbenchmarking.org/result/2004299-NI-RH829042090
https://openbenchmarking.org/result/2004296-NI-RH929042041
https://openbenchmarking.org/result/2004294-NI-RH102904269
https://openbenchmarking.org/result/2004296-NI-RH112904273
https://openbenchmarking.org/result/2004295-NI-RH132904297

Compared RHEL - SELINUX Enabled vs RHEL - SELINUX Disabled
https://openbenchmarking.org/result/2004301-NI-2004295NI92
https://openbenchmarking.org/result/2004305-NI-2004298NI77
https://openbenchmarking.org/result/2004308-NI-2004293NI23
https://openbenchmarking.org/result/2004302-NI-2004293NI66
https://openbenchmarking.org/result/2004307-NI-2004294NI55
https://openbenchmarking.org/result/2004309-NI-2004299NI51
https://openbenchmarking.org/result/2004306-NI-2004296NI51
https://openbenchmarking.org/result/2004309-NI-2004294NI01
https://openbenchmarking.org/result/2004305-NI-2004296NI12
https://openbenchmarking.org/result/2004303-NI-2004295NI01

Debian - SELINUX Enabled
https://openbenchmarking.org/result/2005013-NI-DEB01010562
https://openbenchmarking.org/result/2005015-NI-DB020105219
https://openbenchmarking.org/result/2005012-NI-DEB03010549
https://openbenchmarking.org/result/2005013-NI-DEB04010533
https://openbenchmarking.org/result/2005014-NI-DEB05010570
https://openbenchmarking.org/result/2005016-NI-DB060105223
https://openbenchmarking.org/result/2005015-NI-DEB07010510
https://openbenchmarking.org/result/2005016-NI-DEB08010540
https://openbenchmarking.org/result/2005013-NI-DEB09010587
https://openbenchmarking.org/result/2005018-NI-DEB10010530

Compared Debian - SELINUX Enabled vs Debian - SELINUX Disabled
https://openbenchmarking.org/result/2005025-NI-2005013NI36
https://openbenchmarking.org/result/2005023-NI-2005015NI97
https://openbenchmarking.org/result/2005028-NI-2005012NI95
https://openbenchmarking.org/result/2005024-NI-2005013NI13
https://openbenchmarking.org/result/2005029-NI-2005014NI32
https://openbenchmarking.org/result/2005023-NI-2005016NI51
https://openbenchmarking.org/result/2005026-NI-2005015NI04
https://openbenchmarking.org/result/2005027-NI-2005016NI87
https://openbenchmarking.org/result/2005021-NI-2005013NI48
https://openbenchmarking.org/result/2005025-NI-2005018NI69

Ubuntu - AppArmor Enabled
https://openbenchmarking.org/result/2005037-NI-UBU01030537
https://openbenchmarking.org/result/2005030-NI-UBU02030579
https://openbenchmarking.org/result/2005033-NI-UBU03030517
https://openbenchmarking.org/result/2005033-NI-UBU04030541

https://openbenchmarking.org/result/2005031-NI-UBU05030574
https://openbenchmarking.org/result/2005036-NI-UBU06030579
https://openbenchmarking.org/result/2005030-NI-UBU07030528
https://openbenchmarking.org/result/2005037-NI-UBU08030507
https://openbenchmarking.org/result/2005036-NI-UBU09030568
https://openbenchmarking.org/result/2005030-NI-UBU11030543

Compare Ubuntu - AppArmor Enabled vs Ubuntu - AppArmor Disabled
https://openbenchmarking.org/result/2005047-NI-2005037NI29
https://openbenchmarking.org/result/2005040-NI-2005030NI76
https://openbenchmarking.org/result/2005047-NI-2005033NI09
https://openbenchmarking.org/result/2005044-NI-2005033NI36
https://openbenchmarking.org/result/2005044-NI-2005031NI28
https://openbenchmarking.org/result/2005043-NI-2005036NI19
https://openbenchmarking.org/result/2005042-NI-2005030NI59
https://openbenchmarking.org/result/2005040-NI-2005037NI13
https://openbenchmarking.org/result/2005047-NI-2005036NI20
https://openbenchmarking.org/result/2005044-NI-2005030NI43

Entorn multicore

RHEL - SELINUX Enabled
https://openbenchmarking.org/result/2005098-NI-RHE01090542
https://openbenchmarking.org/result/2005091-NI-RHE02090529
https://openbenchmarking.org/result/2005091-NI-RHE03090528
https://openbenchmarking.org/result/2005098-NI-RHE04090574
https://openbenchmarking.org/result/2005090-NI-RHE05090561
https://openbenchmarking.org/result/2005090-NI-RHE06090515
https://openbenchmarking.org/result/2005096-NI-RHE07090524
https://openbenchmarking.org/result/2005096-NI-RHE08090594
https://openbenchmarking.org/result/2005092-NI-RHE09090519
https://openbenchmarking.org/result/2005091-NI-RHE11090514

Compared RHEL - SELINUX Enabled vs RHEL - SELINUX Disabled
https://openbenchmarking.org/result/2005090-NI-2005098NI68
https://openbenchmarking.org/result/2005090-NI-2005091NI90
https://openbenchmarking.org/result/2005097-NI-2005091NI76
https://openbenchmarking.org/result/2005093-NI-2005098NI25
https://openbenchmarking.org/result/2005097-NI-2005090NI31
https://openbenchmarking.org/result/2005096-NI-2005090NI09
https://openbenchmarking.org/result/2005091-NI-2005096NI03
https://openbenchmarking.org/result/2005096-NI-2005096NI97
https://openbenchmarking.org/result/2005096-NI-2005092NI73
https://openbenchmarking.org/result/2005097-NI-2005091NI31

Debian - SELINUX Enabled
https://openbenchmarking.org/result/2005097-NI-DEB01090565
https://openbenchmarking.org/result/2005098-NI-DEB02090561
https://openbenchmarking.org/result/2005090-NI-DEB03090579
https://openbenchmarking.org/result/2005099-NI-DEB04090526
https://openbenchmarking.org/result/2005096-NI-DEB05090533
https://openbenchmarking.org/result/2005091-NI-DEB06090586
https://openbenchmarking.org/result/2005094-NI-DEB07090575
https://openbenchmarking.org/result/2005099-NI-DEB08090596
https://openbenchmarking.org/result/2005090-NI-DEB09090501
https://openbenchmarking.org/result/2005098-NI-DEB11090585

Compared Debian - SELINUX Enabled vs Debian - SELINUX Disabled
https://openbenchmarking.org/result/2005092-NI-2005097NI03
https://openbenchmarking.org/result/2005099-NI-2005098NI81
https://openbenchmarking.org/result/2005098-NI-2005090NI38
https://openbenchmarking.org/result/2005095-NI-2005099NI86
https://openbenchmarking.org/result/2005095-NI-2005096NI63
https://openbenchmarking.org/result/2005097-NI-2005091NI21
https://openbenchmarking.org/result/2005095-NI-2005094NI74

https://openbenchmarking.org/result/2005095-NI-2005099NI69
https://openbenchmarking.org/result/2005095-NI-2005090NI55
https://openbenchmarking.org/result/2005090-NI-2005098NI82

Ubuntu - AppArmor Enabled
https://openbenchmarking.org/result/2005097-NI-UBU01090587
https://openbenchmarking.org/result/2005092-NI-UBU02090503
https://openbenchmarking.org/result/2005098-NI-UBU03090551
https://openbenchmarking.org/result/2005096-NI-UBU04090528
https://openbenchmarking.org/result/2005092-NI-UBU05090554
https://openbenchmarking.org/result/2005090-NI-UBU06090529
https://openbenchmarking.org/result/2005093-NI-UBU07090538
https://openbenchmarking.org/result/2005098-NI-UBU08090558
https://openbenchmarking.org/result/2005091-NI-UBU09090542
https://openbenchmarking.org/result/2005091-NI-UBU11090577

Compare Ubuntu - AppArmor Enabled vs Ubuntu - AppArmor Disabled
https://openbenchmarking.org/result/2005091-NI-2005097NI65
https://openbenchmarking.org/result/2005094-NI-2005092NI85
https://openbenchmarking.org/result/2005099-NI-2005098NI60
https://openbenchmarking.org/result/2005096-NI-2005096NI08
https://openbenchmarking.org/result/2005090-NI-2005092NI05
https://openbenchmarking.org/result/2005092-NI-2005090NI62
https://openbenchmarking.org/result/2005092-NI-2005093NI38
https://openbenchmarking.org/result/2005094-NI-2005098NI81
https://openbenchmarking.org/result/2005091-NI-2005091NI86
https://openbenchmarking.org/result/2005094-NI-2005091NI64

Bibliografia

- https://www.incibe-cert.es/blog/selinux-y-control-de-acceso-obligatorio
- https://en.wikipedia.org/wiki/Linux_Security_Modules
- https://www.kernel.org/doc/html/latest/security/lsm.html
- https://www.phoronix-test-suite.com/
- https://openbenchmarking.org/
- https://jokeyrhyme.github.io/blog/2013/08/27/1/phoronix_test_suite_on_aws.html
- https://elpuig.xeill.net/Members/vcarceler/articulos/tests-con-phoronix-test-suit-y-openbenchmarking.org
- https://wiki.ubuntu.com/AppArmor
- https://gitlab.com/apparmor/apparmor/-/wikis/Documentation
- https://vidatecno.net/entendiendo-apparmor-en-ubuntulinux/
- https://www.cyberciti.biz/faq/howto-query-components-of-selinux-policy/
- https://www.computernetworkingnotes.com/rhce-study-guide/selinux-explained-with-examples-in-easy-language.html
- https://linuxconfig.org/introduction-to-selinux-concepts-and-management
- https://www.nsa.gov/Portals/70/images/resources/everyone/digital-media-center/publications/research-papers/configuring-selinux-policy-report.pdf