

INSTALAÇÃO DE SERVIÇOS

ADMINISTRAÇÃO E GESTÃO DE REDES E SERVIÇOS

João Sousa, João Veloso e Rui Joaquim

7140199, 7140201,

ÍNDICE

Índice de Figuras	3
Lista de Acrónimos	5
Capítulo 1: Introdução	6
Capítulo 2: Cenário e configuração da Rede	8
Capítulo 3: Configuração de Serviços	10
3.1. Serviços Pressupostos	10
3.1.1 – Serviço DNS	10
3.1.1.1 – Sobre o serviço	10
3.1.1.2 – Funcionamento prático do serviço	10
3.1.1.3 – Configuração do serviço	11
3.1.2 – Serviço NFS	16
3.1.2.1 - Sobre o serviço	16
3.1.2.2 - Funcionamento prático do serviço	16
3.1.2.3 - Configuração do serviço	16
3.1.3 – Serviço NTP	20
3.1.3.1 – Sobre o serviço	20
3.1.3.2 – Funcionamento prático do serviço	20
3.1.3.3 – Configuração do serviço	20
3.1.4 – Serviço SSH	23
3.1.4.1 – Sobre o serviço	23
3.1.4.2 – Funcionamento prático do serviço	23
3.1.4.3 – Configuração do serviço	23
3.2. Serviços Investigados	25
3.2.1 – Configuração do Apache	25
3.2.2 – Configuração do HAProxy	26
3.2.3 – Configuração do PostgreSQL e PHP5	29
3.2.4 – Configuração do Moodle e GIT	32
3.2.5 – Configuração do Webmin	38
Capítulo 4: Scripts	40
Capítulo 5: Conclusão	41
Capítulo 6: Bibliografia	42
Capítulo 7: Anexos	44
7.1 – Serviços Pressupostos	44
7.1.1 – DNS	44
7.1.2 – NTP	45

7.2 – Serviços Estudados	46
7.2.3 Instalação do Moodle	46

ÍNDICE DE FIGURAS

Figura 1 - Esquema de implementação	9
Figura 2 - Configuração Estática de uma Interface de rede	11
Figura 3 - Configuração de servidores de DNS alternativos	12
Figura 4 - Definição do Administrador de domínio	13
Figura 5 - Configuração da Zona AGRSmoodle.pt	14
Figura 6 - Zonas de DNS criadas localmente	15
Figura 7 - Resolução inversa de dns	15
Figura 8 - Filesystems Suportados pela Máquina	17
Figura 9 - Partilhas de NFS	18
Figura 10 - Configuração fstab	19
Figura 11 - atualização pública ntp	22
Figura 12 - Atualização privada ntp	22
Figura 13 - Agendamento de sincronização	22
Figura 14 - Logótipo Apache	25
Figura 15 - Logótipo HAproxy	26
Figura 16 - Arranque automático do serviço haproxy	26
Figura 17 - Configuração failover do haproxy	28
Figura 18 - Logótipo PHP	29
Figura 19 - Logótipo Postgresql	29
Figura 20 - prompt postgresql	29
Figura 21 - Criação de um utilizador na base de dados	30
Figura 22 - Logótipo moodle	32
Figura 23 - Logótipo git	32
Figura 24 - clone ao repositório	32
Figura 25 - "Branches" Disponíveis	33
Figura 26 - Anexação ao "branch"	33
Figura 27 - Verificação do estado do "branch"	34
Figura 28 - Cópia do repositório para pasta local	34
Figura 29 - Interface de instalação do moodle 3.0	35
Figura 30 - Alteração ficheiro config.php	36
Figura 31 - "trust" aos endereços na rede	36
Figura 32 - permissão de ligações externas	36
Figura 33 - Teste final de ligação	37
Figura 34 - Logótipo webmin	38
Figura 35 - Interface de login do webmin	39
Figura 36 - Script de backup	40
Figura 37 - Configuração da zona AGRSmoodle.pt	44
Figura 38 - Configuração da zona inversa	44
Figura 39 - Configuração do serviço ntp	45
Figura 40 - atualização pública ntp	45
Figura 41 - Atualização privada ntp	45
Figura 42 - Interface de instalação do moodle via web	46
Figura 43 - Verificação dos ficheiros necessários 1	46
Figura 44 - Verificação dos ficheiros necessários 2	47

Figura 45 - Análise bem sucedida 1	47
Figura 46 - Análise bem sucedida 2	48
Figura 47 - Configuração de uma conta de administrador do moodle 1	48
Figura 48 - Configuração de uma conta de administrador do moodle 2	49
Figura 49 - Configuração de uma conta de administrador do moodle 3	49
Figura 50 - Configuração do site	49
Figura 51 - Adição de um curso 1	50
Figura 52 - Adição de um curso 2	50
Figura 53 - Teste de cursos	51
Figura 54 - Curso AGRS	51

LISTA DE ACRÓNIMOS

A - Address

AGRS - Administração e Gestão de Redes e Serviços

CNAME - Canonical Name

DHCP - Dynamic Host Configuration Protocol

DNS - Domain Name Service

GPG - GNU Privacy Guard

GUI - Graphical User Interface

IP - Internet Protocol

LTS - Long Time Support

MX - Mail e Xchange

NAS - Network-Attached Storage

NFS - Network File System

NS - Name Server

NTP - Network Time Protocol

PC - Personal Computer

PHP - Hypertext Preprocessor

PTR - Pointer

RAID - Redundant Array of Independent Drives

RAM - Random Access Memory

RO - Read-only

RW - Read and Write

SSH - Secure Shell

TCP - Transmission Control Protocol

TTL - Time To Live

TXT - Text

WWW - World Wide Web

CAPÍTULO 1: INTRODUÇÃO

Este trabalho consiste na implementação de um aglomerado de serviços pressupostos, bem como de serviços investigados, configurados da forma mais interligada possível.

Para tal, foi estabelecido um cenário recorrendo a várias máquinas que desempenham diferentes funções, simulando o funcionamento do sistema informático de uma instituição educacional.

Posto isto, foi realizado um documento que consta de informações acerca dos mesmos serviços, bem como as suas tarefas e os procedimentos de configuração.

Os seguintes motivos justificam as escolhas destes serviços:

A utilização do serviço *Bind* foi fortemente motivada pela documentação fornecida e experiência transmitida pela Unidade Curricular;

A adoção do NFS nas máquinas foi levada a cabo pela simplicidade de configuração e, uma vez mais, pela experiência adquirida na Unidade Curricular;

A integração de uma *Shell* segura, vulgo SSH, é de extrema importância. Nos dias correntes, toda a segurança é pouca, nomeadamente na exposição de informações confidenciais a terceiros. O SSH visa, não só a segurança de comunicações, como também, disponibiliza uma ferramenta de fácil administração em máquinas remotas;

A implementação do serviço de alta disponibilidade, *HAProxy*, foi altamente influenciada por sugestões dos docentes da Unidade Curricular, tendo sido despertado o interesse dos formandos por este tipo de serviço;

A escolha do *Moodle 3.0* recaiu no facto de ser um dos mais populares serviços de ensino na atualidade. A instalação do mesmo, via *git*, permite que obtenhamos o mais recente repositório de ficheiros criados pela Moodle.org, sendo facilitada a sua instalação e configuração em sistemas *Linux/Unix*.

Para além de estar referido na escolha do tema para este trabalho, a escolha assente na base de dados *PostgreSQL* permitiu alargar os nossos horizontes a nível do conhecimento de bases de dados. Anteriormente já tínhamos trabalhado com *MySQL* e apesar de ser uma

das bases de dados recomendadas pelo *Moodle*, optámos pelo *PostgreSQL* para desenvolver a nossa intelectualidade.

A nível de interfaces gráficas para facilitar a manutenção do sistema, o *Webmin* foi a nossa escolha. A sua instalação é muito simples e ao acrescentarmos o repositório do *Webmin* ao utilitário *APT*, possibilitou, um acesso ainda mais rápido e fluído ao pacote de instalação.

CAPÍTULO 2: CENÁRIO E CONFIGURAÇÃO DA REDE

Este trabalho de investigação visa implementar um cenário virtual que simula um estabelecimento de ensino, no qual se verifica uma típica configuração de serviços na infraestrutura de rede que garante o alojamento de páginas *Web* (*Apache*) pertencentes à instituição, bem como a integração de uma plataforma de ensino (*Moodle*) para facilitar o acesso a documentos e informações sobre unidades curriculares, com registo de informações, utilizando software de gestão de bases de dados (*PostgreSQL*).

Recorrendo ao conceito de virtualização por *software* (execução de máquinas virtuais a partir de uma física, através da partilha de recursos de *hardware*), foram configuradas máquinas virtuais, replicando o funcionamento de máquinas totalmente independentes.

Descrição das máquinas configuradas:

Neste projeto, existe um total de quatro máquinas em funcionamento: duas delas, formam um *cluster* (conjunto de nós, com serviços replicados, que funcionam como uma única máquina) para assegurar o funcionamento do *Moodle*.

Outra das máquinas, será o nó central do *cluster* tendo a implementação de serviços pressupostos ao funcionamento de uma rede privada, assim como, um serviço que garante a Alta Disponibilidade (*Failover*).

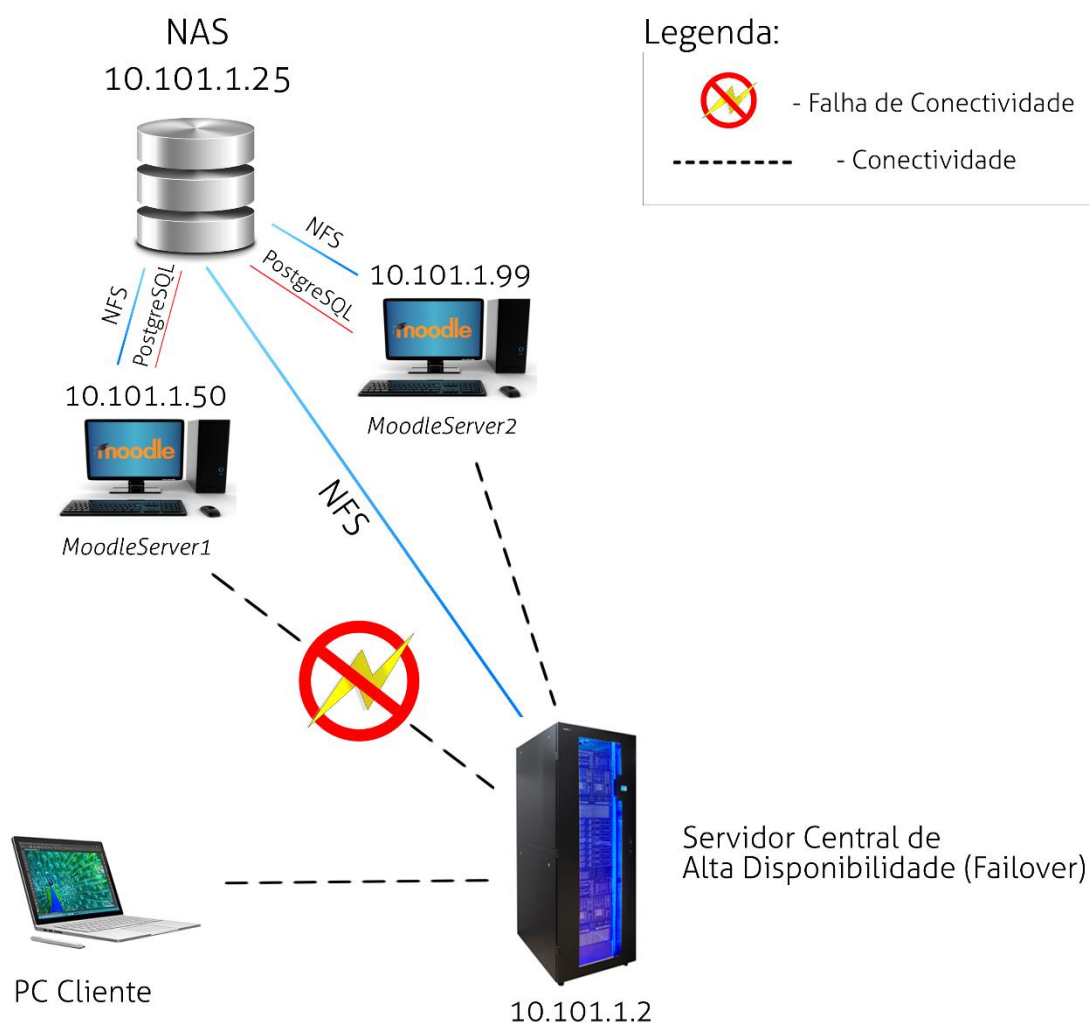


FIGURA 1 - ESQUEMA DE IMPLEMENTAÇÃO

Todas as máquinas virtuais utilizam o Sistema Operativo *Ubuntu Server 14.04.3 LTS* e possuem 500 MB de memória RAM alocada em cada máquina.

Como se pode verificar, no diagrama acima representado, um cliente tem de passar primeiramente pelo Servidor Central do *cluster* que tem o encargo de redirecionar o pedido para o servidor *Web* respetivo. Posteriormente, este servir-se-á das bases de dados alojadas na NAS, estando também responsável pelo alojamento de cópias de segurança de ficheiros de configuração críticos dos serviços de rede. A NAS é, deste modo, a central por onde todas as informações vitais estão guardadas.

O grande objetivo deste trabalho de investigação é, no fundo, de projetar um acesso transparente por parte dos clientes ao *cluster* mesmo em caso de falha do servidor principal, isto é, omitir possíveis alterações entre os servidores que atendem os pedidos *Web* (na prática, a página devolvida será sempre semelhante), transmitindo a perspetiva de um ambiente direto e sempre disponível (com valores aceitáveis de *uptime*).

CAPÍTULO 3: CONFIGURAÇÃO DE SERVIÇOS

3.1. Serviços Pressupostos

3.1.1 – SERVIÇO DNS

3.1.1.1 – SOBRE O SERVIÇO

Ao efetuar ligação a qualquer página da Internet, ou a configurar dispositivos numa rede de média ou larga escala, é impensável a necessidade de memorizar todos os endereços de recursos com que se quer comunicar. Esta dificuldade gerou necessidade em desenvolver o DNS (*Domain Name Service* - Serviço de Nomes de Domínio), que consiste num serviço de resolução de nomes em endereços IP implementado em servidor. Este pode também funcionar de forma inversa, em que a partir de um endereço IP se faz uma pesquisa por nomes associados (chamada de pesquisa inversa - *reverse lookup*).

Neste trabalho de investigação, foi implementado um domínio local que permite um acesso facilitado à plataforma *Moodle* do sistema, denominado “**agrsmoodle.pt**”, com respetivas associações de nomes de máquinas aos respetivos endereços IP.

3.1.1.2 – FUNCIONAMENTO PRÁTICO DO SERVIÇO

O DNS consiste na atribuição de um nome ao endereço IP de uma máquina (por exemplo, é possível aceder alternativamente à página da *Google* Portugal “www.google.pt”, através da ligação direta ao endereço de um dos servidores da *Google* que responde a este pedido de acesso, como o 216.58.210.99).

Um servidor de DNS pode também em qualquer momento passar simultaneamente por cliente, visto que conforme disponibiliza resoluções de nomes, pode também fazer consultas de nomes ou endereços, questionando outros servidores externos. Exemplos de aplicações de cliente DNS são o **Dig** (em *Linux*), **nslookup** (*Windows* e *Linux*), ou **qualquer navegador da Internet**.

É possível verificar no *Ubuntu* a que servidor DNS uma máquina está a recorrer, através da visualização do ficheiro “**/etc/resolv.conf**”, no campo “*nameserver*” (servidor de nomes).

3.1.1.3 – CONFIGURAÇÃO DO SERVIÇO

De forma a configurar um servidor DNS estático, será necessário instalar *software* apropriado. Em sistemas operativos baseados em *Unix*, por norma recorre-se ao *Bind*, sendo gratuito e a solução mais popular para este fim. A sua versão atual é a 9.10.3, lançada a 16 de setembro de 2015.

Para instalar o *Bind*, recorre-se ao comando:

```
sudo apt-get install bind9
```

Antes de iniciar a sua configuração, é recomendável a definição de endereços estáticos nas máquinas da rede interna, editando o ficheiro das *interfaces* de rede, localizado em “*/etc/network/interfaces*”:

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0

#iface eth0 inet dhcp

iface eth0 inet static
    address 10.101.1.2
    netmask 255.255.255.0
    gateway 10.101.1.1
    dns-nameservers 127.0.0.1
    dns-search agrsmoodle.pt
```

FIGURA 2 - CONFIGURAÇÃO ESTÁTICA DE UMA INTERFACE DE REDE

- O campo “**address**” deverá ser alterado conforme o endereço IP pretendido;
- No campo “**netmask**” deverá ser inserida a máscara de rede correspondente à rede;
- No campo “**gateway**” deverá ser inserido o endereço IP do *gateway* da rede;
- No campo “**dns-nameservers**” deve ser colocado 127.0.0.1 no caso do servidor DNS (este endereço que indica a própria máquina, o *localhost*), e o endereço IP do servidor DNS nas restantes máquinas;
- No campo “**dns-search**” insere-se o domínio privado implementado com o *Bind*, neste caso “*agrsmoodle.pt*”.

Para finalizar este passo, deve-se reiniciar o serviço de rede e forçar uma nova atribuição de endereço IP, recorrendo aos comandos:

- **`sudo /etc/init.d/networking restart`**
- **`sudo ifdown eth0`**
- **`sudo ifup eth0`**

Inicia-se então à configuração do *Bind*. Os seus ficheiros de configuração estão localizados na diretoria **`/etc/bind`**, que dispõe dos seguintes ficheiros:

- **db.** - Ficheiros predefinidos nomeadamente destinados à configuração de zonas, como o `db.0` ou `db.255` que parametrizam uma zona de *broadcast* privada;
- **named.** – Definem os principais parâmetros de funcionamento do *bind*. Distribui-se por:
 - **named.conf** - ficheiro que indica que ficheiros de configuração o *bind* deverá utilizar;
 - **named.conf.default-zones** - contém as zonas predefinidas do *bind*;
 - **named.conf.options** - destinado principalmente à definição de *forwarders*, isto é, encaminhadores alternativos que o servidor deverá utilizar para respostas não-autoritárias (pedidos de resolução externos a outros servidores na *Internet*, caso o servidor não consiga fazer a resolução de um nome)
 - **named.conf.local** - utilizado para parametrizar zonas locais.

Configuração de named.conf.options

Neste ficheiro deve ser retirado o comentário à estrutura da diretiva `"forwarders"`, e adicionar servidores DNS alternativos, um por linha, que serão utilizados para respostas não-autoritárias.

Neste cenário, optou-se por definir os servidores DNS da *Google*, resultando no seguinte formato:

```
forwarders {  
    8.8.8.8;  
    8.8.4.4;  
};
```

FIGURA 3 - CONFIGURAÇÃO DE
SERVIDORES DE DNS ALTERNATIVOS

Criação do domínio local agrsmoodle.pt

De seguida, será necessário criar o domínio local. Este procedimento foi aprofundado para implementar o domínio **agrsmoodle.pt**, que permite um acesso facilitado por parte de um cliente na rede à plataforma *Moodle* sem necessidade de memorizar o endereço IP do servidor *Web*.

Para tal, é recomendável **criar primeiro uma diretoria adicional** dentro da pasta do *bind*, denominada por exemplo "zones", onde serão alojados os ficheiros de configuração do domínio local da zona, utilizando o comando:

```
sudo mkdir /etc/bind/zones
```

Para criar a zona **agrsmoodle.pt**, utiliza-se como modelo de estrutura o ficheiro predefinido "*db.local*", copiando-o para a pasta "zones" com o novo nome:

```
sudo cp /etc/bind/db.local /etc/bind/zones/db.agrsmoodle.pt
```

Configuração da zona de DNS (db.agrsmoodle.pt)

É no ficheiro acabado de criar que será de seguida realizada a configuração de nomes das máquinas da zona.

Na **segunda** linha será necessário substituir os campos "*localhost*." pelo nome da máquina do domínio que configura o DNS, e o campo seguinte por um possível endereço de contacto de *e-mail* do administrador do domínio, resultando numa linha como:

```
@      IN      SOA      cluster.agrsmoodle.pt. admin.agrsmoodle.pt. (
```

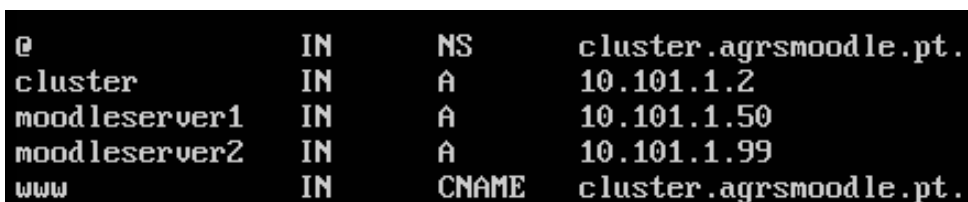
FIGURA 4 - DEFINIÇÃO DO ADMINISTRADOR DE DOMÍNIO

Alguns parâmetros gerais da zona que podem ser configurados são:

- **Serial**- versão do ficheiro, geralmente alterado conforme mudanças nos valores do ficheiro;
- **Refresh** - tempo entre sincronizações da zona;
- **Retry** - em caso de falha, tempo de espera até tentar novas sincronizações;
- **Expire** - tempo total permitido para pedidos;
- **Negative cache TTL** - TTL (*Time to Live* – número máximo de saltos entre nós até descartar um pacote), para configuração de servidores DNS de *cache*.

No final do documento, colocam-se todas as máquinas e respetivos parâmetros, atribuindo as máquinas aos encargos e nomes correspondentes. Pode ser utilizada a seguinte nomenclatura:

- **NS** - Servidor que responde pela zona do *domínio*;
- **A** - IPv4 *Address* (usado para destinar nomes a endereços IPv4);
- **AAAA** - IPv6 *Address* (usado para destinar nomes a endereços IPv6);
- **CNAME** - Usado na atribuição de nomes alternativos a máquinas do domínio;
- **MX** - Servidores responsáveis pelo intercâmbio de *mail*;
- **PTR** - *Pointer* (apontador utilizado nas resoluções inversas de DNS);



```
@           IN      NS       cluster.agrsmoodle.pt.
cluster     IN      A        10.101.1.2
moodleserver1  IN    A        10.101.1.50
moodleserver2  IN    A        10.101.1.99
www         IN      CNAME    cluster.agrsmoodle.pt.
```

FIGURA 5 - CONFIGURAÇÃO DA ZONA AGRSMOODLE.PT

(Na subcategoria 7.1.1. será possível verificar a versão final implementada no cenário.)

Configuração de named.conf.local

Neste ficheiro de configuração será incluída a zona acabada de criar bem como a zona de resolução inversa que será criada a seguir (na zona inversa os octetos do endereço IP são escritos do fim para o início):

```
zone "agrs Moodle.pt" {
    type master;
    file "/etc/bind/zones/db.agrs Moodle.pt";
};

zone "1.101.10.in-addr.arpa" {
    type master;
    file "/etc/bind/zones/db.10.101.1";
};
```

FIGURA 6 - ZONAS DE DNS CRIADAS LOCALMENTE

Configuração da zona inversa (db.10.101.1)

A zona inversa é definida da mesma forma que a anterior, exceto que estará localizada num ficheiro à parte, utilizando como modelo o ficheiro “db.127”. Copia-se, alterando o nome para os octetos do endereço IP da rede:

```
sudo cp /etc/bind/db.127 /etc/bind/zones/db.10.101.1
```

Segue-se o mesmo procedimento que no passo “Configuração da zona de DNS”, exceto que na primeira coluna se coloca apenas o octeto do *host* das máquinas, e se usa a nomenclatura “PTR”.

@	IN	NS	cluster.agrs Moodle.pt.
2	IN	PTR	cluster.agrs Moodle.pt.
50	IN	PTR	moodleserver1.agrs Moodle.pt.
99	IN	PTR	moodleserver2.agrs Moodle.pt.

FIGURA 7 - RESOLUÇÃO INVERSA DE DNS

Para finalizar, reinicia-se o serviço *bind* (“***sudo service bind9 restart***”) nas máquinas afetadas. Pode ser testado o seu devido funcionamento, utilizando os seguintes comandos:

- “**Dig** (endereço IP do servidor)”
- “**Dig -x** (nome da máquina do servidor)”

Uma correta configuração devolverá o nome da máquina no primeiro caso e o endereço IP respetivo ao nome inserido no segundo caso.

3.1.2 – SERVIÇO NFS

3.1.2.1 - SOBRE O SERVIÇO

O NFS (*Network File System* – Sistema de Ficheiros Remoto) é um serviço implementado para sistemas baseados em *Unix* e destinado à partilha de partições através da rede. É destacado pela sua simplicidade na configuração, fiabilidade, e pelo facto de ser totalmente gratuito. A sua versão mais recente é o NFSv4.1, sendo estável e mais robusta.

Existem outros tipos semelhantes de *Software*, adotados consoante diferentes tipos de necessidades, como por exemplo o *Samba*, que já suporta características de partilha entre diferentes tipos de sistemas operativos, notavelmente de *Linux* para *Windows*.

Este serviço foi implementado em todas as máquinas da rede, em que há transferência de *backups* de ficheiros de configuração de serviços dos servidores *Moodle* para a máquina de alojamento em rede (NAS) e também da máquina central do *cluster* para a mesma.

3.1.2.2 - FUNCIONAMENTO PRÁTICO DO SERVIÇO

O NFS consiste em realizar a partilha de uma partição ou diretoria do disco rígido de uma máquina-servidor com outras máquinas, sendo destinado ao acesso via rede, em que o espaço é apenas ocupado no servidor. Visualmente, e na prática, o cliente acede a esta diretoria e interage com a mesma (conforme as permissões atribuídas) como se o recurso pertencesse ao seu próprio sistema operativo.

3.1.2.3 - CONFIGURAÇÃO DO SERVIÇO

Configuração do ambiente em servidor

De forma a configurar um servidor NFS, será necessário em primeiro lugar instalar o pacote completo através do comando:

```
sudo apt-get install nfs-kernel-server
```

Para confirmar a correta instalação, recorre-se ao comando “**cat /proc/filesystems**”, verificando na lista resultante se os *filesystems* “**nfs**”, “**nfs4**” e “**nfsd**” estão presentes.

```
nodev    pstore
nodev    mqueue
nodev    rpc_pipefs
nodev    nfs
nodev    nfs4
nodev    nfsd
```

FIGURA 8 - FILESYSTEMS SUPORTADOS PELA MÁQUINA

Antes de iniciar as configurações de partilhas, caso ainda não exista a(s) diretoria(s) a partilhar, será necessário criá-la(s). Neste cenário, como se utiliza a NAS para alojar os *backups* de ficheiros de configuração das restantes três máquinas, foi planeada a criação de **uma diretoria principal** na raiz, contendo **três subdiretorias**, uma para cada das máquinas:

```
sudo mkdir -p /backups/m1
```

```
sudo mkdir /backups/m2
```

```
sudo mkdir /backups/c
```

Uma vez criadas, estas diretorias podem então ser definidas no ficheiro de diretorias exportadas pelo NFS, localizado em “**/etc/exports**”, que serão posteriormente montadas a partir das máquinas-cliente. Deverá ser indicada uma diretoria por linha, assim como o endereço IP com quem a partilhar, junto com alguns parâmetros, tais como:

- **Ro** - Restringe as permissões do conteúdo da diretoria para apenas leitura;
- **Rw** - Restringe as permissões do conteúdo da diretoria para leitura e escrita;
- **Sync** – Otimiza as sincronizações com as máquinas a fazer partilha;
- **No_wdelay** – Evita atrasos na escrita do disco. Requer a opção “*sync*” ativada;
- **No_root_squash** – Mantém permissões *root* ao exportar o conteúdo da diretoria para outra máquina.
- **No_subtree_check** – Desativa a validação de pesquisas em subdiretorias exportadas dentro de diretorias não exportadas pelo servidor NFS.

As seguintes configurações foram aplicadas no trabalho de investigação:

```
/backups/m1    10.101.1.50(rw,sync,no_wdelay,no_subtree_check,no_root_squash)
/backups/m2    10.101.1.99(rw,sync,no_wdelay,no_subtree_check,no_root_squash)
/backups/c     10.101.1.2(rw,sync,no_wdelay,no_subtree_check,no_root_squash)
```

FIGURA 9 - PARTILHAS DE NFS

Para finalizar, atualizam-se as configurações utilizando o comando “*exportfs*” e reinicia-se o serviço NFS:

```
sudo exportfs -ra
```

```
sudo service nfs-kernel-server restart
```

É possível verificar as diretorias exportadas, pelo comando:

```
sudo exportfs -v
```

Configuração do cliente NFS

Para configurar um cliente NFS, será necessário instalar o pacote básico para garantir o acesso a um servidor:

```
sudo apt-get install nfs-common
```

(Ou, em vez do anterior, pode ser instalado de novo o pacote “*nfs-kernel-server*”, para também preparar a máquina para uma possível futura implementação como servidor)

Uma vez mais, pode ser efetuada a confirmação do suporte aos *filesystems* do NFS usando o comando “*cat /proc/filesystems*”, verificando se estes estão presentes.

Daqui, resta criar uma diretoria-destino para receber a partilha:

```
sudo mkdir /nfs
```

E, por fim, adicionar uma linha referente ao endereço IP e a partição exportada no ficheiro de montagem automática de *filesystems* aquando do arranque da máquina, localizado em “**/etc/fstab**”, junto com as configurações-padrão da partilha:

```
10.101.1.25:/backups/m1      /nfs    nfs     defaults    0      2
```

FIGURA 10 - CONFIGURAÇÃO FSTAB

Para garantir o acesso à diretoria, é necessário reiniciar a máquina, verificando posteriormente se esta se encontra presente no caminho definido.

Daqui, testa-se este serviço criando ficheiros de teste na diretoria, confirmando se estes se localizam em ambas as máquinas.

3.1.3 – SERVIÇO NTP

3.1.3.1 – SOBRE O SERVIÇO

O NTP (*Network Time Protocol*) consiste num protocolo desenvolvido para realizar sincronizações com alto nível de precisão da data e hora de um sistema a partir de um servidor. Este é realizado através da rede, sendo possível realizar pedidos a servidores configurados internamente (servidores privados), ou através da *Internet* (questionando servidores públicos).

O cenário implementado consiste na configuração de um servidor NTP privado na máquina central do *cluster*, que possui também funcionalidades de cliente, respondendo a pedidos deste tipo às outras máquinas existentes na rede e questionando esporadicamente dois servidores públicos, de forma a manter uma constante sincronização.

3.1.3.2 – FUNCIONAMENTO PRÁTICO DO SERVIÇO

Na prática, existem **clientes** e **servidores NTP**: um cliente **apenas questiona** um servidor de forma a receber estas informações da forma mais precisa possível, realizando assim a sua própria sincronização, enquanto que um servidor as **disponibiliza** a outras máquinas, podendo **também opcionalmente recorrer** a funcionalidades de cliente, onde este acede a uma *pool* (conjuntos de servidores) pública.

3.1.3.3 – CONFIGURAÇÃO DO SERVIÇO

Configuração do servidor NTP

Em primeiro lugar, será necessário adquirir o pacote e instalar o NTP, de forma a configurar o ficheiro do serviço:

```
sudo apt-get install ntp
```

Procede-se então à configuração do NTP, cujo ficheiro está localizado em **`“/etc/ntp.conf”`**.

Aqui, no campo dos servidores NTP a utilizar serão acrescentadas duas linhas correspondentes aos servidores da *Internet* que a máquina utilizará para se sincronizar (é recomendável utilizar no mínimo dois servidores, de forma a obter um valor redundante e mais correto), que serão:

- ✓ **server 0.pt.ntp.pool.org**
- ✓ **server 2.pt.ntp.pool.org**

Outras duas linhas a acrescentar correspondem à configuração da máquina como servidor NTP, sendo estas:

- ✓ **server 127.127.1.0** - endereço interno que aponta para a própria máquina para se disponibilizar como servidor privado para outras máquinas na rede;
- ✓ **fudge 127.127.1.0 stratum 10** - garante que o servidor da máquina local se situa num nível de servidores NTP (*stratum*) menos prioritário que os públicos, para evitar conflitos.

Para terminar, comenta-se a linha **`“server ntp.ubuntu.com”`** (acrescentando um símbolo **`“#”`** atrás do seu começo) de forma a forçar o funcionamento do NTP com apenas os três servidores anteriormente especificados (nos anexos será possível observar a configuração implementada no trabalho de investigação).

As alterações só serão aplicadas reiniciando o *daemon* (processo executado com o arranque da máquina que garante o funcionamento do serviço) do NTP, utilizando o comando:

```
sudo /etc/init.d/ntp restart
```

Para testar o funcionamento do serviço, pode-se optar por forçar uma sincronização a um dos servidores escolhidos, recorrendo ao comando “**ntpdate**” (com opção “-u” para atualizar durante a execução do processo NTP):

```
cluster@moodlecluster:~$ sudo ntpdate -u 0.pt.pool.ntp.org
8 Jan 00:45:41 ntpdate[1589]: step time server 193.136.164.1 offset 4.221639 sec
cluster@moodlecluster:~$ _
```

FIGURA 11 - ATUALIZAÇÃO PÚBLICA NTP

Esta mensagem indica que houve atualização com sucesso.

Configuração do cliente NTP

Não existe qualquer configuração específica para efetuar pedidos como cliente - apenas será necessário realizar testes de ligação ao servidor NTP, tal como no passo anterior, recorrendo ao comando “**ntpdate**”:

```
cluster@server1:~$ sudo ntpdate -u 192.168.233.128
[sudo] password for cluster:
8 Jan 00:54:30 ntpdate[1397]: adjust time server 192.168.233.128 offset -0.080622 sec
cluster@server1:~$
```

FIGURA 12 - ATUALIZAÇÃO PRIVADA NTP

Tendo realizado uma correta sincronização, pode-se **definir um agendamento automático** da execução deste comando nas máquinas de forma personalizada, através da edição do ficheiro para tal destinado, o **Cron**, com o comando “**sudo crontab -e**”

```
# m h dom mon dow   command
0 3 * * 0 ntpdate -u 192.168.233.128
```

FIGURA 13 - AGENDAMENTO DE SINCRONIZAÇÃO

Adiciona-se a última linha, que neste caso permitirá **realizar o comando de sincronização NTP todos os domingos, às 3 horas da manhã**.

(**Nota:** a configuração de um servidor NTP privado gerou problemas e falhas de sincronização, cuja investigação e *troubleshooting* levou à conclusão que a infraestrutura da rede onde se realizaram testes deste serviço impunha limitações ao seu funcionamento,

tendo sido necessário recorrer a um acesso externo configurado com NAT, o que originou a utilização de uma diferente gama de endereços IP para verificar o funcionamento

do serviço. Teoricamente, o NTP foi corretamente projetado e aplicado, porém a nível prático, poderão existir resultados inconclusivos consoante a rede onde se testa.)

3.1.4 – SERVIÇO SSH

3.1.4.1 – SOBRE O SERVIÇO

O SSH (*Secure Shell*) consiste num serviço que permite efetuar acessos a máquinas remotas por meio de um túnel cifrado, garantindo uma segurança reforçada contra possíveis ataques de furto ou manipulação de informações trocadas. Este serviço trabalha na porta 22 TCP.

3.1.4.2 – FUNCIONAMENTO PRÁTICO DO SERVIÇO

Este serviço destina-se a efetuar ligações remotas, possibilitando trocas de dados de uma forma cifrada, protegendo-os. É uma potente ferramenta que facilita a administração de máquinas à distância, simultaneamente garantindo segurança nesta tarefa.

3.1.4.3 – CONFIGURAÇÃO DO SERVIÇO

Para realizar a sua instalação recorre-se ao comando:

```
sudo apt-get install openssh-server
```

A partir daqui o serviço encontrar-se-á a funcionar, mas é recomendado que se proceda à sua configuração. Para isso teremos de aceder ao ficheiro de configuração “*/etc/ssh/sshd_config*”, onde será possível definir alguns parâmetros personalizados como:

- **AllowUsers** **@xxx.xxx.xxx.xxx* – para restringir o acesso a apenas algumas máquinas (substituir o último campo pelo endereço IP da máquina a permitir acesso; repetir o comando quantas vezes necessário, um por linha);
- **LoginGraceTime** **30** – para definir o tempo máximo permitido para fazer *login* (neste caso 30 segundos);
- **PermitRootLogin** **no** – para negar o acesso ao utilizador *root*.

Estas configurações serão aplicadas assim que o serviço for reiniciado com o comando:

```
sudo service ssh restart
```

Acesso de clientes SSH

No lado do cliente, existem duas principais formas de aceder a uma outra máquina por SSH: por linha de comandos, ou por método gráfico (exemplo: *software putty*).

Para aceder à *shell* por comando, utilizamos como exemplo:

- **ssh 'cluster@10.101.1.2'**

Aparecerá um aviso a indicar que a autenticidade da máquina-destino não pôde ser verificada, perguntando se pretende continuar mesmo assim; selecciona-se "yes". Com esta confirmação, será transmitida uma mensagem de aviso confirmando que a máquina indicada foi adicionada à lista de *hosts* conhecidos. Será pedida então a *password* do utilizador que se pretende conectar, que, após inserida corretamente, terá acesso à máquina.

Para aceder por GUI, por exemplo em *windows*, podemos utilizar o **putty**. Apenas temos de inserir o endereço IP destino no campo "**Host Name(or IP address)**" e confirmar que se está a utilizar a porta 22.

Após clicar em "Open" será pedido o nome de utilizador e a respetiva palavra-passe, estabelecendo assim ligação.

(**Nota:** Se existir uma configuração por acesso restrito à máquina, é necessário certificar que o endereço IP foi previamente adicionado à lista de utilizadores permitidos.)

APACHE

O Apache é um *Software* destinado à implementação de um servidor onde seja possível criar, alojar e disponibilizar páginas *Web*. A sua estabilidade, fiabilidade e simplicidade, bem como a sua distribuição totalmente gratuita e de livre edição para qualquer programador (*Open-source Software*) tornou este a mais popular escolha a quem pretenda configurar um servidor *Web*.

3.2. Serviços Investigados

3.2.1 – Configuração do Apache



FIGURA 14 - LOGÓTIPO APACHE

Para proceder à sua instalação, será necessário recorrer ao comando:

```
sudo apt-get install apache2
```

Este permitirá adquirir os pacotes do *software* bem como todos os seus requisitos automaticamente, garantindo imediatamente o funcionamento do serviço. A instalação ocupará cerca de 5 MB de espaço no disco rígido.

A única configuração restante a realizar para este serviço será a alteração da diretoria raiz de acesso aos ficheiros da página principal do servidor, acedendo ao seu ficheiro de configuração (*"/etc/apache2/sites-available/000-default.conf"*), alterando a linha:

- **DocumentRoot /var/www/html**

Para:

- **DocumentRoot /var/www/html/moodle**

Termina-se, reiniciando o serviço *Apache*:

```
sudo service apache2 restart
```


HAPROXY

O *HAProxy* (*High Availability Proxy* – *Proxy* de alta disponibilidade) consiste num serviço gratuito implementado a partir de uma máquina central, ocupando o encargo de encaminhar o tráfego recebido de pedidos de máquinas entre um dado número de servidores.

Na prática, conforme as configurações estipuladas, este serviço caracteriza-se por assegurar balanceamento de carga, sendo capaz de alternar acessos e comunicação entre diferentes servidores interligados, ou garantir alta disponibilidade, transferindo o funcionamento de um serviço para outra máquina em caso de falha com a principal, melhorando o *uptime* dum sistema.

3.2.2 – CONFIGURAÇÃO DO HAPROXY



FIGURA 15 - LOGÓTIPO HAPROXY

Em primeiro lugar, será necessário adquirir o pacote e instalar o *HAProxy*, utilizando o comando:

```
sudo apt-get install haproxy
```

Antes de realizar posteriores configurações, é recomendável assegurar de imediato que este será iniciado automaticamente com o arranque da máquina, acedendo ao ficheiro “*/etc/default/haproxy*”, onde se altera o valor de “*ENABLED*” para ‘1’:

```
# Set ENABLED to 1 if you want the init script to start haproxy.
ENABLED=1
# Add extra flags here.
#EXTRA_OPTS="-de -m 16"
```

FIGURA 16 - ARRANQUE AUTOMÁTICO DO SERVIÇO HAPROXY

De seguida, procede-se à configuração do serviço, no ficheiro “*/etc/haproxy/haproxy.cfg*”

O cenário implementado no trabalho de investigação visa adotar e assegurar um funcionamento contínuo do serviço *Web* que fornece a plataforma *Moodle* (método *failover*), garantindo o *uptime* de um sistema em caso de falha ou negação de serviço no servidor principal.

Assim sendo, de uma forma geral foram mantidas as configurações-padrão do ficheiro, destacando-se a adição das seguintes linhas:

Temporização de ligações:

Os seguintes valores podem ser alterados consoante necessidade do administrador. Estes apenas são aplicáveis quando se configura o *HAProxy* em modo de balanceamento de carga.

- ✓ **Timeout connect** – tempo máximo para um cliente se tentar ligar ao servidor, até alternar de servidor que atende os pedidos;
- ✓ **Timeout client** – tempo máximo de inatividade por parte do cliente (troca de dados) até alternar o servidor atendedor dos pedidos;
- ✓ **Timeout server** – tempo máximo de inatividade por parte do servidor (troca de dados) até alternar o servidor atendedor dos pedidos;

Configurações do funcionamento do serviço

- ✓ **Frontend** – esta configuração define como os pedidos são encaminhados para os servidores definidos. Aqui deve-se integrar o porto do serviço pelo qual os servidores configurados estarão à escuta (no caso de servidores *Web*, o porto 80);
- ✓ **Backend** – esta diretiva permite estabelecer as configurações dos pedidos enviados aos servidores. Nesta secção, define-se principalmente o método de balanceamento de tráfego estabelecido, e a indicação das máquinas com que o *HAProxy* vai funcionar.

Efetuuou-se a seguinte configuração neste ficheiro, de forma a permitir o funcionamento *Failover*.

```
timeout connect 5000
timeout client 3000
timeout server 50000

frontend ApacheFrontend
    bind *:80
    default_backend ApacheBackend

backend ApacheBackend
    balance roundrobin
    mode http
    server apache1 10.101.1.50:80 check inter 1s fall 3 rise 1
    server apache2 10.101.1.99:80 check backup
```

FIGURA 17 - CONFIGURAÇÃO FAILOVER DO HAPROXY

- ✓ A opção “**balance roundrobin**” permite a alternância de pedidos entre as diferentes máquinas;
- ✓ A opção “**check**” confirma a comunicação (ou não) com o servidor listado;
- ✓ A opção “**inter**” define o intervalo entre cada tentativa de comunicação com o servidor (caso esta falhe);
- ✓ A opção “**fall**” consiste no número de tentativas de comunicação com o servidor (caso esta falhe uma vez), até o classificar como desligado e transferir o serviço para o servidor seguinte;
- ✓ A opção “**rise**” consiste no número de tentativas de comunicação com o primeiro servidor, caso haja recuperação de serviços, até a reclassificar como operacional;
- ✓ A opção “**backup**” é adicionada ao servidor destinado à recuperação de serviços, indicando que esta só será utilizada caso a primeira falhe.

O serviço estará pronto a utilizar, sendo apenas necessário reiniciar o serviço para aplicar as alterações:

```
sudo service haproxy restart
```

POSTGRESQL E PHP5

PostgreSQL é uma poderosa ferramenta *open source* de gestão de base de dados. É uma das muitas alternativas ao *MySQL*.

Já o *PHP*, na sua versão 5 é uma linguagem de *scripting* utilizada em aplicações que utilizam bases de dados como *MediaWiki*, *Facebook*, *Wordpress*, *Joomla*, entre outras.

3.2.3 – CONFIGURAÇÃO DO POSTGRESQL E PHP5



FIGURA 19 - LOGÓTIPO POSTGRESQL



FIGURA 18 - LOGÓTIPO PHP

De modo a possibilitar uma correta configuração do *Moodle*, o servidor necessitará de ter instalado a linguagem de *scripting* *PHP* versão 5 e a base de dados *PostgreSQL*.

O sistema operativo necessitará das seguintes dependências de *PHP5*:

```
sudo apt-get install php5-fpm php-apc php5-curl
php5-gd php5-xmlrpc php5-intl
```

Já a instalação do *PostgreSQL* necessitará destas:

```
sudo apt-get install postgresql postgresql-contrib
php5-pgsql
```

Após a instalação, iniciamos a base de dados com o comando:

```
sudo -u postgres psql
```

```
moodleadmin@MoodleServer1:~$ sudo -u postgres psql
psql (9.3.10)
Type "help" for help.

postgres=#
```

FIGURA 20 - PROMPT POSTGRESQL

O Administrador deve agora criar uma conta de utilizador dentro da *prompt* do *postgres*:

```
CREATE USER moodleadadmin WITH PASSWORD
'1';
```

```
postgres=# CREATE USER moodleadadmin WITH PASSWORD '1';
CREATE ROLE
postgres=#
```

FIGURA 21 - CRIAÇÃO DE UM UTILIZADOR NA BASE DE DADOS

É criada agora a base de dados para o utilizador 'moodleadadmin'

```
sudo -u postgres created -E utf8 -O moodleadadmin moodle
```

É protegido o acesso à base de dados com uma password de *admin*, neste caso '1'.

```
sudo -u postgres psql template1
# ALTER USER postgres WITH PASSWORD '1';

# \q
```

Após a criação bem-sucedida do utilizador e da base de dados, o administrador pode agora sair da *prompt* do *postgres* com o comando `\q` + tecla ENTER.

Instalamos ainda algumas dependências restantes para o serviço:

```
sudo apt-get install graphviz aspell php5-pspell php5-curl php5-gd php5-
intl php5-mysql php5-xmlrpc php5-ldap clamav
```

(A instalação do módulo `php5-mysql` é opcional).

Reinicia-se a base de dados:

```
sudo /etc/init.d/postgresql-x.x restart
```

(é inserido o valor da versão atual do *postgresql* que esteja a ser utilizada no lugar dos 'x' aqui representados).

Com a configuração inicial do *PostgreSQL*, do PHP5 e com a criação de um utilizador dá-se por concluída a instalação destes serviços.

Prossegue-se então à instalação do serviço de controlo *Git* para obter a versão mais recente do *Moodle* a partir dos repositórios oficiais no subcapítulo seguinte.

GIT E MOODLE

Git é um sistema de controlo de versões focado na rapidez e na integração de dados. É um *software Open Source*.

O *Moodle* é um serviço *Web* criado a pensar nas necessidades dos estudantes e das entidades de Ensino. Pretende disponibilizar uma plataforma onde o conteúdo pode ser partilhado entre professores e alunos. O *Moodle* possibilita ainda a capacidade de ensino à distância, sendo possível a disponibilização de aulas e de testes na plataforma.

3.2.4 – CONFIGURAÇÃO DO MOODLE E GIT



FIGURA 23 - LOGÓTIPO GIT

FIGURA 22 - LOGÓTIPO MOODLE

Para iniciar a instalação do software *Git*, procedemos à seguinte inserção na linha de comandos:

```
sudo apt-get install git-core
```

Após a instalação, prossegue-se à configuração do repositório local do administrador. Ao utilizar o *Git* está a facilitar o seu trabalho futuramente no que diz respeito à forma como atualiza a aplicação do *Moodle*.

Da localização onde está na *bash*, dirija-se à diretoria */opt* através do comando:

```
cd /opt
```

Faça *download* do código fonte do *Moodle* e do seu índice através de uma clonagem aos repositórios oficiais do *Moodle*:

```
sudo git clone git://git.moodle.org/moodle.git
```

```
moodleadmin@MoodleServer1:~$ cd /opt
moodleadmin@MoodleServer1:/opt$ sudo git clone git://git.moodle.org/moodle.git
Cloning into 'moodle'...
remote: Counting objects: 858904, done.
remote: Compressing objects: 100% (200846/200846), done.
Receiving objects: 0% (424/858904), 92.00 KiB | 12.00 KiB/s
```

FIGURA 24 - CLONE AO REPOSITÓRIO

Mude de diretoria para a pasta onde transferiu o *Moodle*:

```
cd moodle
```

Faça uma recolha de todos os “*branches*” disponíveis no repositório:

```
sudo git branch -a
```

```
moodleadmin@MoodleServer1:/opt$ cd moodle
moodleadmin@MoodleServer1:/opt/moodle$ sudo git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/MOODLE_13_STABLE
remotes/origin/MOODLE_14_STABLE
remotes/origin/MOODLE_15_STABLE
remotes/origin/MOODLE_16_STABLE
remotes/origin/MOODLE_17_STABLE
remotes/origin/MOODLE_18_STABLE
remotes/origin/MOODLE_19_STABLE
remotes/origin/MOODLE_20_STABLE
remotes/origin/MOODLE_21_STABLE
remotes/origin/MOODLE_22_STABLE
remotes/origin/MOODLE_23_STABLE
remotes/origin/MOODLE_24_STABLE
remotes/origin/MOODLE_25_STABLE
remotes/origin/MOODLE_26_STABLE
remotes/origin/MOODLE_27_STABLE
remotes/origin/MOODLE_28_STABLE
remotes/origin/MOODLE_29_STABLE
remotes/origin/MOODLE_30_STABLE
remotes/origin/master
moodleadmin@MoodleServer1:/opt/moodle$ _
```

FIGURA 25 - "BRANCHES" DISPONÍVEIS

```
sudo git branch --track MOODLE_30_STABLE
origin/MOODLE_30_STABLE
```

Diga ao *Git* que “*branch*” pretende utilizar (Neste caso foi utilizada a versão 3.0 do Moodle):

```
moodleadmin@MoodleServer1:/opt/moodle$ sudo git branch --track MOODLE_30_STABLE origin/MOODLE_30_STABLE
Branch MOODLE_30_STABLE set up to track remote branch MOODLE_30_STABLE from origin.
moodleadmin@MoodleServer1:/opt/moodle$ _
```

FIGURA 26 - ANEXAÇÃO AO "BRANCH"

Verifique qual é a mais recente versão do *Moodle*:

```
sudo git checkout MOODLE_30_STABLE
```

```
sudo cp -R /opt/moodle /var/www/html/
```

```
sudo mkdir /var/moodledata
```

```
sudo chown -R www-data /var/moodledata
```

```
sudo chmod -R 777 /var/moodledata
```

```
sudo chmod -R 0755 /var/www/html/moodle
```

```
moodleadmin@MoodleServer1:/opt/moodle$ sudo git checkout MOODLE_30_STABLE
Switched to branch 'MOODLE_30_STABLE'
Your branch is up-to-date with 'origin/MOODLE_30_STABLE'.
moodleadmin@MoodleServer1:/opt/moodle$ _
```

FIGURA 27 - VERIFICAÇÃO DO ESTADO DO "BRANCH"

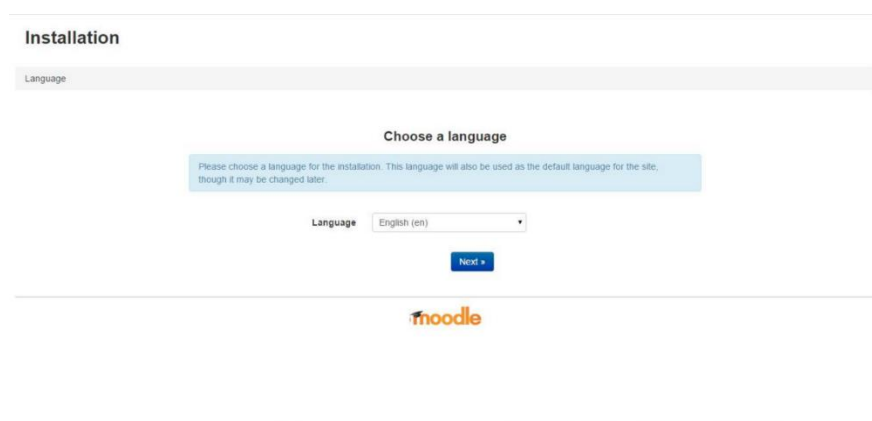
Copie o repositório local para a diretoria /var/www/html/

```
moodleadmin@MoodleServer1:/opt/moodle$ sudo cp -R /opt/moodle /var/www/html/
moodleadmin@MoodleServer1:/opt/moodle$ sudo mkdir /var/moodledata
moodleadmin@MoodleServer1:/opt/moodle$ sudo chown -R www-data /var/moodledata
moodleadmin@MoodleServer1:/opt/moodle$ sudo chmod -R 777 /var/moodledata
moodleadmin@MoodleServer1:/opt/moodle$ sudo chmod -R 0755 /var/www/html/moodle
moodleadmin@MoodleServer1:/opt/moodle$
```

FIGURA 28 - CÓPIA DO REPOSITÓRIO PARA PASTA LOCAL

Explicação sucinta: Como se configurou um repositório local no passo anterior, este irá ser copiado para o *webroot* após quaisquer *updates* ou alterações que tenham sido feitos. Ter o repositório local fora do *webroot*, tal como foi feito na diretoria /opt, permite que os upgrades ao *Moodle* sejam realizados de uma forma mais eficiente. Por exemplo, quando se instalam *plugins*, normalmente, é transferido o *plugin* e este é copiado para o repositório local do *Moodle*. Possivelmente, terá depois de editar o ficheiro localizado na diretoria /opt/moodle/.git/info/exclude. Dentro desse ficheiro, serão configuradas as pastas que devem ser excluídas pelo *git* quando este faz um “pull” aos *updates* do repositório assim que é efetuado o comando “*sudo git pull*”.

Por fim, através de um browser é possível finalizar a instalação do *Moodle* 3.0 através do endereço <http://localhost/moodle> (em que o *localhost* é o IP definido na máquina do servidor). Após toda a instalação, é necessário apontar os servidores *Moodle* para o domínio local criado, alterando a linha “*\$CFG->wwwroot*” do ficheiro “*/etc/www/html/moodle/config.php*” para o valor “*http://www.agrsmoodle.pt*”.



Para uma instalação detalhada, verifique o subcapítulo 7.2.3. (Anexos).

FIGURA 29 - INTERFACE DE INSTALAÇÃO DO MOODLE 3.0

Como anteriormente explicado, este cenário foi implementado com um total de quatro servidores: dois suportam a plataforma *moodle*; um armazena backups e as bases de dados utilizadas pelo *moodle*, e uma máquina para garantir alta disponibilidade de serviços.

Em cada um dos dois servidores de *moodle*, após a instalação dos respetivos serviços, foi modificado o ficheiro `"/var/www/html/moodle/config.php"` de forma a poderem aceder às bases de dados remotas:

- `$CFG->dbhost = '10.101.1.25';`
- `$CFG->dbuser = 'moodleadmin';`
- `$CFG->dbpass = '1';`

Foi alterada ainda a linha que determina o endereço de acesso ao *moodle*, de forma a ser universal entre os dois servidores:

```

$CFG->dbtype      = 'pgsql';
$CFG->dblibrary    = 'native';
$CFG->dbhost       = '10.101.1.25';
$CFG->dbname       = 'moodle';
$CFG->dbuser       = 'moodleadmin';
$CFG->dbpass       = '1';
$CFG->prefix       = 'mdl_';
$CFG->dboptions    = array (
    'dbpersist' => 0,
    'dbport'    => '',
    'dbsocket'  => '',
);

$CFG->wwwroot      = 'http://www.agrsmoodle.pt';
$CFG->dataroot      = '/var/www/moodledata';
$CFG->admin         = 'admin';

```

FIGURA 30 - ALTERAÇÃO FICHEIRO CONFIG.PHP

De seguida, de forma a que a máquina servidor de base de dados possa aceitar pedidos de acesso, foi adicionada a seguinte linha no ficheiro "/etc/postgresql/9.3/main/pg_hba.conf":

- **host all all 10.101.1.0/24 trust**

```

# replication privilege.
#local   replication    postgres
#host     replication    postgres        127.0.
#host     replication    postgres        ::1/12
host all all 10.101.1.0/24 trust

```

FIGURA 31 - "TRUST" AOS ENDEREÇOS NA REDE

No ficheiro /etc/postgresql/9.3/main/postgresql.conf retirou-se a marca de comentário e alterou-se para:

- **listen_addresses='*'**

```

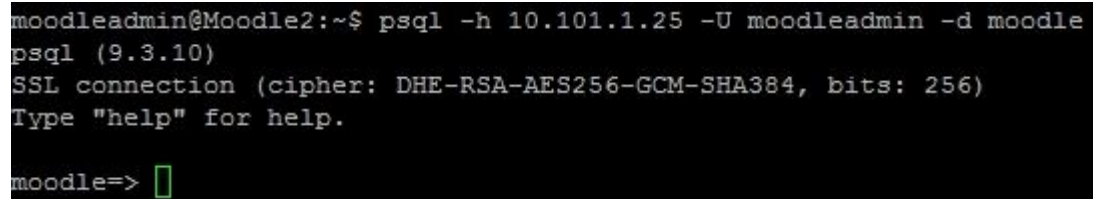
#-----
# - Connection Settings -
listen_addresses = '*'           # w
port = 5432

```

FIGURA 32 - PERMISSÃO DE LIGAÇÕES EXTERNAS

Finalmente, após reiniciar o serviço *postgresql*, os servidores *moodle* terão ligação remota às bases de dados armazenadas pelo servidor NAS:

/etc/init.d/postgresql restart

A terminal window with a black background and white text. The prompt is 'moodleadmin@Moodle2:~\$'. The command entered is 'psql -h 10.101.1.25 -U moodleadmin -d moodle'. The output shows 'psql (9.3.10)', 'SSL connection (cipher: DHE-RSA-AES256-GCM-SHA384, bits: 256)', and 'Type "help" for help.'. The prompt changes to 'moodle=>' with a green cursor.

```
moodleadmin@Moodle2:~$ psql -h 10.101.1.25 -U moodleadmin -d moodle
psql (9.3.10)
SSL connection (cipher: DHE-RSA-AES256-GCM-SHA384, bits: 256)
Type "help" for help.

moodle=> █
```

FIGURA 33 - TESTE FINAL DE LIGAÇÃO

WEBMIN

Webmin é uma ferramenta de administração gráfica *web-based* que facilita a administração de um sistema operativo *Linux/Unix* através de um simples *browser*.

3.2.5 – CONFIGURAÇÃO DO WEBMIN



FIGURA 34 - LOGÓTIPO WEBMIN

A instalação do *Webmin* neste projeto foi realizada através de repositórios oficiais utilizando o comando *apt*. Para tal, foi necessário adicionar os repositórios à lista de repositórios do sistema operativo.

```
sudo nano /etc/apt/sources.list
```

Adicionaram-se as seguintes linhas no fim do ficheiro:

```
deb
http://download.webmin.com/download/repository
sarge contrib
deb
http://webmin.mirror.somersettechsolutions.co.uk/re
pository sarge contrib
```

Ainda dentro do editor nano, pressiona-se CTRL + X, Y e ENTER para salvar o ficheiro com as modificações.

É necessário importar uma chave GPG pela qual este repositório se cinge. Para tal são utilizados os seguintes comandos:

```
wget http://www.webmin.com/jcameron-key.asc
sudo apt-key add jcameron-key.asc
```

Após a configuração dos repositórios, a instalação do *webmin* torna-se mais simples. A sua localização predefinida é `/usr/share/webmin`.

```
sudo apt-get update
```

```
sudo apt-get install webmin
```

Com a colocação destes simples comandos, a instalação do *webmin* está concluída.

O acesso à sua *interface* é feito por *browser*, acedendo ao site <https://localhost:10000> sendo que *localhost* é o IP da máquina e 10000 é o porto em que o serviço do *webmin* está à escuta.

As credenciais de acesso são as mesmas que estão configuradas para o sistema operativo.

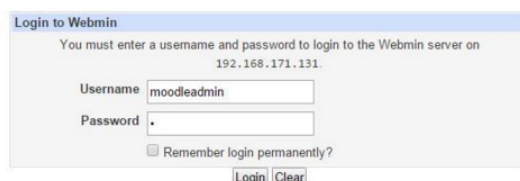


FIGURA 35 - INTERFACE DE LOGIN DO WEBMIN

CAPÍTULO 4: SCRIPTS

(Ver ficheiro **Script Backup do Moodle.txt** na pasta scripts para uma visualização mais legível)

Foi desenvolvido um *script* em *Perl* para realizar uma cópia de segurança de ficheiros de configuração de serviços críticos. Este *script* utiliza uma lista auxiliar em .txt para indicar as pastas para *backup*.

```

1  #!/usr/bin/perl
2  use strict;
3  use warnings;
4  use Sys::Syslog;
5
6
7  my $destino = "/nfs/";
8  chomp(my $dow = `date +%w`); #
9  chomp(my $today = `date +%A`);
10 chomp(my $hostname = `hostname -s`);
11 my $nome_file = "$hostname-$today.tgz";
12 chomp(my $whoami = `whoami`);
13
14 chomp(my $check = `ls /nfs | wc -l`);
15
16
17 if(@ARGV != 1){
18     print STDERR ("Erro de sintaxe: [ $0 <settings>]\n");
19     exit(1);
20 }else{
21     if(!-e "$ARGV[0]"){
22         print ("O ficheiro \"$ARGV[0]\" não existe\n");
23         openlog($whoami, "ndelay,pid", "local3");
24         syslog("err", "Ficheiro \"$ARGV[0]\" não existente\n");
25         closelog();
26         exit(2);
27     }else{
28         openlog($whoami, "ndelay,pid", "local3");
29         syslog("info", "Iniciado backup\n");
30         closelog();
31
32         open(FILE, "<", $ARGV[0]); # abrir o ficheiro no argumento
33         chomp (my @backup_files = <FILE>);
34         close(FILE);
35
36         if($dow == 0 || $check < 1){ # Backup completo se for Domingo OU se a pasta estiver vazia
37             print("Backup Completo\n");
38             system("tar -cpzvf $destino/$nome_file @backup_files");
39
40             openlog($whoami, "ndelay,pid", "local3");
41             syslog("info", "Backup completo finalizado\n");
42             closelog();
43             print("Backup completo finalizado\n");
44         }else{
45             print("Backup incremental\n");
46             system("find @backup_files -mtime -1 | xargs tar -cpzvf $destino/$nome_file");# backup incremental
47
48             openlog($whoami, "ndelay,pid", "local3");
49             syslog("info", "Backup incremental finalizado\n");
50             closelog();
51             print("Backup incremental finalizado\n");
52         }
53     }
54 }

```

FIGURA 36 - SCRIPT DE BACKUP

CAPÍTULO 5: CONCLUSÃO

Dos objetivos que traçámos ao longo do período de concretização deste projeto, debruçámo-nos sobre diversos problemas, nomeadamente a nível da integridade das bases de dados *PostgreSQL*.

A instalação dos serviços essenciais foi realizada de forma bem-sucedida, sem grandes percalços, com uma especial exceção para o NTP, que apesar de estar bem configurado, não nos foi possível verificar a sua resolução nos computadores Cliente devido aos bloqueios impostos pelo IPLeiria.

No que toca aos restantes serviços, tentámos implementar sincronização nos dois servidores *Moodle* através do utilitário *rSync*. No entanto, após algum esforço, verificámos que essa não seria a melhor prática e com os consequentes problemas que tivemos a níveis de permissões, decidimos abandonar essa ideia e investigar a alternativa de criar um sistema de NFS na tentativa de recriar a prática que é realizada nos dias de hoje, de possibilitar um *array* de discos à parte de todo o restante equipamento informático.

Após sucessivas intervenções no nosso *Moodle*, verificámos que ao modificar o conteúdo das páginas numa máquina (Exemplo: MoodleServer1), a máquina MoodleServer2, ou vice-versa, não recebiam corretamente a sincronização das bases de dados *PostgreSQL* e chegava até a corromper os dados da base de dados do *Moodle*, sendo que o acesso aos dados era feito diretamente nos discos da máquina virtual que criámos para fazer de NAS.

Preferimos, portanto, adotar ainda outra medida de acesso às bases do *PostgreSQL*, fazendo uma replicação das bases de dados da máquina MoodleServer1 e MoodleServer2 através do *HAProxy*, mas o que podia funcionar na teoria não se conseguiu adaptar na prática.

Até à data de escrita deste relatório, estamos de momento a identificar outras soluções para implementar a sincronização das bases de dados.

CAPÍTULO 6: BIBLIOGRAFIA

Serviço DNS:

"Ficha 7 - DNS" Disponível em: ead.ipleiria.pt/ucs201516; Autores: Carlos Antunes, Leonel Santos, Gustavo Reis, Miguel Frade, Joana Costa, Mário Antunes; Disciplina: AGRS; Guião laboratorial 07_dns.pdf; Acedido em 28 de dezembro de 2015

Serviço NFS:

"Ficha 5 - NFS e Samba" Disponível em: ead.ipleiria.pt/ucs201516; Autores: Mário Antunes, Carlos Antunes, Leonel Santos, Miguel Frade, Joana Costa, Nuno Veiga; Disciplina: AGRS; Guião laboratorial 05_nfs_samba.pdf; Acedido em 27 de dezembro de 2015

Serviço NTP:

"HOWTO: Set Up an NTP Server" Disponível em:
<http://ubuntuforums.org/showthread.php?t=862620>; Acedido em 23 de dezembro de 2015

Serviço SSH:

"Introduction" Disponível em:
<https://help.ubuntu.com/community/SSH/OpenSSH/Configuring>; Acedido em 4 de janeiro de 2016

Serviço HAProxy:

"Apache Load Balancing With HAProxy" Disponível em: <http://dasunhegoda.com/apache-load-balancing-haproxy/659/>; Autor: Dasun Hegoda; Acedido em 27 de dezembro de 2015

"HAproxy configuration and Load balancing" Disponível em:
<https://www.youtube.com/watch?v=L6U0PcESQ4Y>; Autor: Azizullah Parsa; Acedido em 28 de dezembro de 2015

"HAProxy switch only in case of server down" Disponível em:

<http://serverfault.com/questions/473445/haproxy-switch-only-in-case-of-server-down>; Autor: Joe; Acedido em 5 de Janeiro de 2016

Serviço Moodle:

"Step-by-step Installation Guide for Ubuntu" Disponível em:

https://docs.moodle.org/26/en/Step-by-step_Installation_Guide_for_Ubuntu; Acedido em 8 de dezembro de 2015

"Step-by-step Installation Guide for Ubuntu" Disponível em:

https://docs.moodle.org/30/en/Step-by-step_Installation_Guide_for_Ubuntu; Acedido em 8 de dezembro de 2015

Serviço PostgreSQL:

"How Do I Enable remote access to PostgreSQL database server?" Disponível em:

<http://www.cyberciti.biz/tips/postgres-allow-remote-access-tcp-connection.html#>; Autor: Vivek Gite; Acedido em 5 de janeiro de 2016

"PostgreSQL" Disponível em: <https://docs.moodle.org/30/en/PostgreSQL>; Acedido em 17 de dezembro de 2015

"How To Install Moodle via git with Postgres, Nginx and PHP on an Ubuntu 12.04 VPS"

Disponível em: <https://www.digitalocean.com/community/tutorials/how-to-install-moodle-via-git-with-postgres-nginx-and-php-on-an-ubuntu-12-04-vps>; Acedido em 18 de dezembro de 2015

"How to configure postgresql for the first time?", Disponível em:

<http://stackoverflow.com/questions/1471571/how-to-configure-postgresql-for-the-first-time>; Autor: Rohit Banga; Acedido em 18 de dezembro de 2015

Serviço Webmin:

"How to Install Webmin Web Hosting Control Panel on Ubuntu, Debian & LinuxMint",

Disponível em: <http://tecadmin.net/install-webmin-in-ubuntu-and-debian/#>; Autor: Rahul; Acedido em 23 de dezembro de 2015

CAPÍTULO 7: ANEXOS

7.1 – Serviços Pressupostos

7.1.1 – DNS

```

$TTL      604800
@         IN      SOA      cluster.agrsmoodle.pt. admin.agrsmoodle.pt. (
                                4          ; Serial
                                604800     ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )   ; Negative Cache TTL
;

@         IN      NS       cluster.agrsmoodle.pt.
cluster   IN      A        10.101.1.2
moodleserver1 IN    A      10.101.1.50
moodleserver2 IN    A      10.101.1.99
www       IN      CNAME    cluster.agrsmoodle.pt.

```

FIGURA 37 - CONFIGURAÇÃO DA ZONA AGRSMOODLE.PT

```

$TTL      604800
@         IN      SOA      cluster.agrsmoodle.pt. admin.agrsmoodle.pt. (
                                3          ; Serial
                                604800     ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )   ; Negative Cache TTL
;

@         IN      NS       cluster.agrsmoodle.pt.
2         IN      PTR      cluster.agrsmoodle.pt.
50        IN      PTR      moodleserver1.agrsmoodle.pt.
99        IN      PTR      moodleserver2.agrsmoodle.pt.

```

FIGURA 38 - CONFIGURAÇÃO DA ZONA INVERSA

7.1.2 – NTP

```
# /etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help

driftfile /var/lib/ntp/ntp.drift

# Enable this if you want statistics to be logged.
#statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# Specify one or more NTP servers.

# Use servers from the NTP Pool Project. Approved by Ubuntu Technical Board
# on 2011-02-08 (LP: #104525). See http://www.pool.ntp.org/join.html for
# more information.

server 0.pt.pool.ntp.org
server 2.pt.pool.ntp.org
server 127.127.1.0
fudge 127.127.1.0 stratum 10

# Use Ubuntu's ntp server as a fallback.
#server ntp.ubuntu.com
```

FIGURA 39 - CONFIGURAÇÃO DO SERVIÇO NTP

```
cluster@moodlecluster:~$ sudo ntpdate -u 0.pt.pool.ntp.org
8 Jan 00:45:41 ntpdate[1589]: step time server 193.136.164.1 offset 4.221639 sec
cluster@moodlecluster:~$ _
```

FIGURA 40 - ATUALIZAÇÃO PÚBLICA NTP

```
cluster@server1:~$ sudo ntpdate -u 192.168.233.128
[sudo] password for cluster:
8 Jan 00:54:30 ntpdate[1397]: adjust time server 192.168.233.128 offset -0.080622 sec
cluster@server1:~$
```

FIGURA 41 - ATUALIZAÇÃO PRIVADA NTP

7.2 – Serviços Estudados

7.2.3 INSTALAÇÃO DO MOODLE

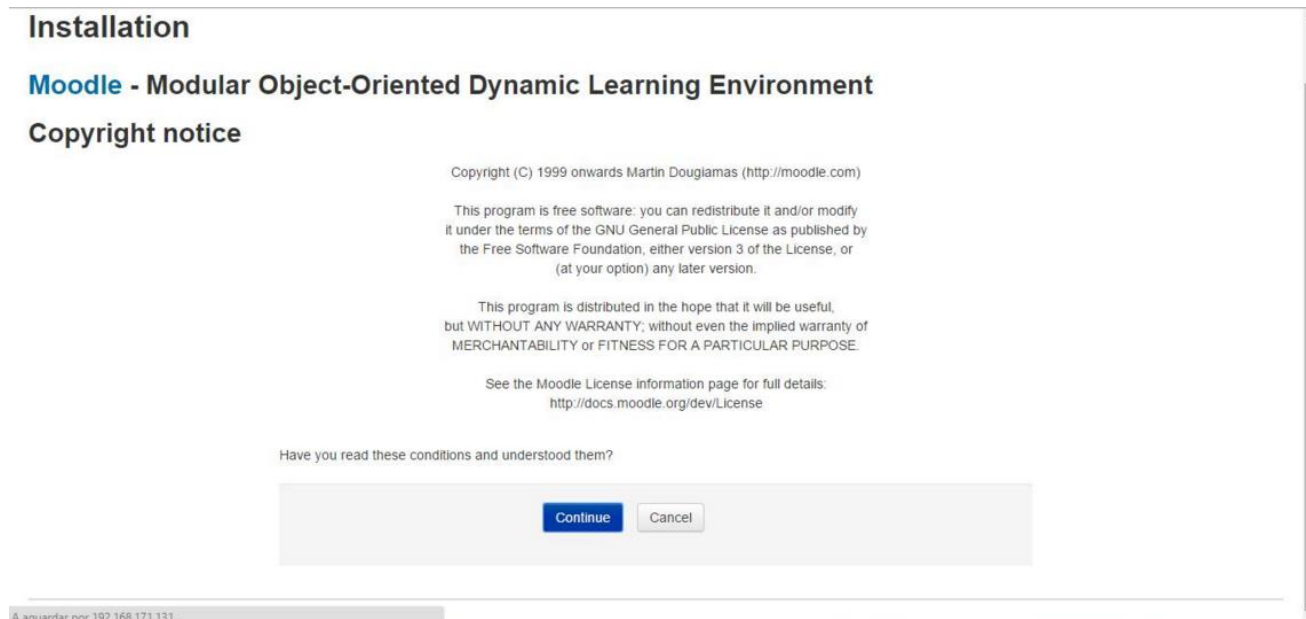


FIGURA 42 - INTERFACE DE INSTALAÇÃO DO MOODLE VIA WEB

Installation - Moodle 3.0.1 (Build: 20151221)

Moodle 3.0.1 (Build: 20151221)

For information about this version of Moodle, please see the online [Release Notes](#)

Server checks

Name	Information	Report	Plugin	Status
unicode		① must be installed and enabled		OK
database	postgres (9.3.10)	① version 9.1 is required and you are running 9.3.10		OK
php		① version 5.4.4 is required and you are running 5.5.9.1.4.14		OK
pcreunicode		① should be installed and enabled for best results		OK
php_extension	iconv	① must be installed and enabled		OK
php_extension	mbstring	① should be installed and enabled for best results		OK
php_extension	curl	① must be installed and enabled		OK
php_extension	openssl	① should be installed and enabled for best results		OK
php_extension	tokenizer	① should be installed and enabled for best results		OK
php_extension	xmlrpc	① should be installed and enabled for best results		OK
php_extension	soap	① should be installed and enabled for best results		OK
php_extension	ctype	① must be installed and enabled		OK
php_extension	zip	① must be installed and enabled		OK

FIGURA 43 - VERIFICAÇÃO DOS FICHEIROS NECESSÁRIOS 1

php_extension	openssl	① should be installed and enabled for best results	OK
php_extension	tokenizer	① should be installed and enabled for best results	OK
php_extension	xmlrpc	① should be installed and enabled for best results	OK
php_extension	soap	① should be installed and enabled for best results	OK
php_extension	ctype	① must be installed and enabled	OK
php_extension	zip	① must be installed and enabled	OK
php_extension	zlib	① must be installed and enabled	OK
php_extension	gd	① must be installed and enabled	OK
php_extension	simplexml	① must be installed and enabled	OK
php_extension	spl	① must be installed and enabled	OK
php_extension	pcr	① must be installed and enabled	OK
php_extension	dom	① must be installed and enabled	OK
php_extension	xml	① must be installed and enabled	OK
php_extension	intl	① should be installed and enabled for best results	OK
php_extension	json	① must be installed and enabled	OK
php_extension	hash	① must be installed and enabled	OK
php_setting	memory_limit	① recommended setting detected	OK
php_setting	file_uploads	① recommended setting detected	OK
php_setting	opcache.enable	① recommended setting detected	OK

Your server environment meets all minimum requirements.

Continue

FIGURA 44 - VERIFICAÇÃO DOS FICHEIROS NECESSÁRIOS 2

Installation
System
Success
availability_completion
Success
availability_date
Success
availability_grade
Success

FIGURA 45 - ANÁLISE BEM SUCEDIDA 1

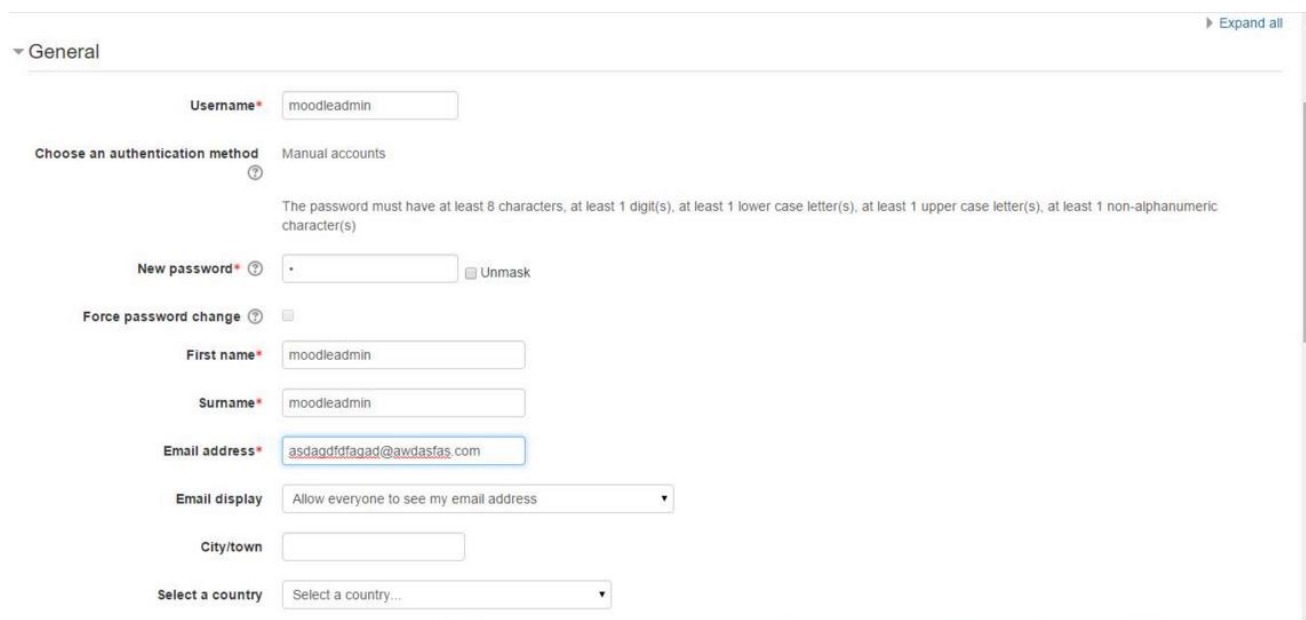


The screenshot displays a vertical list of four components with their installation status:

- tinymce_wrap**: Success
- logstore_database**: Success
- logstore_legacy**: Success
- logstore_standard**: Success

A "Continue" button is located at the bottom of the list.

FIGURA 46 - ANÁLISE BEM SUCEDIDA 2



The screenshot shows the "General" tab of the Moodle administrator account configuration form. The fields are as follows:

- Username***: moodleadadmin
- Choose an authentication method**: Manual accounts (with a help icon)
- New password***: (masked with dots) and an "Unmask" checkbox.
- Force password change**: (checkbox is unchecked, with a help icon)
- First name***: moodleadadmin
- Surname***: moodleadadmin
- Email address***: asdagdfdfagad@awdasfas.com
- Email display**: Allow everyone to see my email address (dropdown menu)
- City/town**: (empty text field)
- Select a country**: Select a country... (dropdown menu)

An "Expand all" link is visible in the top right corner of the form.

FIGURA 47 - CONFIGURAÇÃO DE UMA CONTA DE ADMINISTRADOR DO MOODLE 1

Surname*	<input type="text" value="moodleadmin"/>
Email address*	<input type="text" value="asdagtdfdagad@awdasfas.com"/>
Email display	<input type="text" value="Hide my email address from everyone"/>
City/town	<input type="text"/>
Select a country	<input type="text" value="Portugal"/>
Timezone	<input type="text" value="Server timezone (Europe/Paris)"/>
Description ⓘ	<div><div></div></div>
	<input type="text" value="HTML format"/>

FIGURA 48 - CONFIGURAÇÃO DE UMA CONTA DE ADMINISTRADOR DO MOODLE 2

This summary can be displayed on the front page using the course/site summary block.

New settings - Location settings

Default timezone

timezone

Europe/Lisbon

Default: Europe/Paris

This is the default timezone for displaying dates - each user can override this setting in their profile. Cron tasks and other server settings are specified in this timezone. You should change the setting if it shows as "Invalid timezone"

New settings - Manage authentication

Self registration

registerauth

Disable

Default: Disable

If an authentication plugin, such as email-based self-registration, is selected, then it enables potential users to register themselves and create accounts. This results in the possibility of spammers creating accounts in order to use forum posts, blog entries etc. for spam. To avoid this risk, self-registration should be disabled or limited by *Allowed email domains* setting.

Save changes

FIGURA 49 - CONFIGURAÇÃO DE UMA CONTA DE ADMINISTRADOR DO MOODLE 3

Installation

New settings - Front page settings

Full site name

Short name for site (eg single word)

Front page summary




FIGURA 50 - CONFIGURAÇÃO DO SITE

AGRS moodleadadmin moodleadadmin

Dashboard

NAVIGATION

- Dashboard
- Site home
- Site pages
- Courses

ADMINISTRATION

- Site administration

Search

COURSE OVERVIEW

No course information to show.

PRIVATE FILES

No files available

Manage private files...

ONLINE USERS

(last 5 minutes)

moodleadmin moodleadadmin

LATEST BADGES

You have no badges to display

CALENDAR

December 2015

FIGURA 51 - ADIÇÃO DE UM CURSO 1

AGRS moodleadadmin

Dashboard Site administration Courses Manage courses and categories Add a new course

NAVIGATION

- Dashboard
- Site home
- Site pages
- Courses

ADMINISTRATION

- Category: Miscellaneous
 - Manage this category
 - Edit this category
 - Add a subcategory
 - Assign roles
 - Permissions
 - Check permissions
 - Cohorts
 - Filters
 - Restore course
- Site administration
 - Notifications
 - Registration
 - Advanced features
 - Users

Add a new course

Expand all

General

Course full name*

Course short name*

Course category Miscellaneous

Visible Show

Course start date 24 December 2015

Course ID number

Description

Course summary

FIGURA 52 - ADIÇÃO DE UM CURSO 2

The screenshot shows the Moodle AGRS interface. The top header displays 'AGRS' on the left and 'moodle admin' with a user profile icon on the right. The main heading is 'Teste de cursos'. Below it, a breadcrumb trail reads 'Dashboard > Courses > TDC'. A 'Turn editing on' button is in the top right. The left sidebar contains two sections: 'NAVIGATION' with links to Dashboard, Site home, Site pages, Current course (TDC), Participants, Badges, and Courses; and 'ADMINISTRATION' with links to Course administration, Turn editing on, Edit settings, Users, Filters, Reports, and Grades. The main content area features a 'News forum' icon and a list of course periods: 24 December - 30 December, 31 December - 6 January, 7 January - 13 January, 14 January - 20 January, and 21 January - 27 January. On the right, there are four widgets: 'SEARCH FORUMS' with a search box and 'Go' button; 'LATEST NEWS' with a link to 'Add a new topic...' and a note that no news has been posted yet; 'UPCOMING EVENTS' with a note that there are no upcoming events and links to 'Go to calendar...' and 'New event...'; and 'RECENT ACTIVITY'.

FIGURA 53 - TESTE DE CURSOS

The screenshot shows the Moodle AGRS interface for the 'Administração e Gestão de Redes e Sistemas' course. The top header displays 'AGRS' on the left and 'moodle admin' with a user profile icon on the right. The main heading is 'Administração e Gestão de Redes e Sistemas'. Below it, a breadcrumb trail reads 'Dashboard > Site home > Site pages > Courses'. A 'Turn editing on' button is in the top right. The left sidebar contains two sections: 'NAVIGATION' with links to Dashboard, Site home, Site pages, and Courses; and 'ADMINISTRATION' with links to Front page settings, Turn editing on, Edit settings, Users, Filters, Reports, Backup, Restore, Question bank, and Site administration. The main content area features an 'Available courses' section with a link to 'Teste de cursos' and a note: 'Este curso serve apenas para teste do sistema.' On the right, there is a 'CALENDAR' widget showing the month of December 2015 with a grid of dates from 1 to 31.

FIGURA 54 - CURSO AGRS