



# **Orientação Técnica (OTC)**

Red Hat JBoss EAP 7

EQUIPE:	
<b>Elaboração: DEAT/DIAS</b>	<b>Luís Felipe Vital Abreu Fonseca</b>
<b>Responsável: DEPS/DITF</b>	<b>Álvaro Teixeira Gonçalves</b>
<b>Aprovação: DEPS/DITF</b>	<b>Marcelo André Ferreira Silva</b>

#### HISTÓRICO DE REVISÕES

DATA	VERSÃO	DESCRIÇÃO	AUTOR
03/08/16	1	Criação	<b>Luís Felipe Vital Abreu Fonseca</b>
25/08/16	1	Ajuste - mod_slotmem	<b>Luís Felipe Vital Abreu Fonseca</b>
30/08/16	1	Revisão	<b>Álvaro T. Gonçalves</b>
29/12/16	1	Adequação do nome do documento	<b>Álvaro T. Gonçalves</b>

## SUMÁRIO

1. OBJETIVO.....	5
2. CONTEXTUALIZAÇÃO.....	5
3. VISÃO GERAL.....	5
4. CONHECIMENTOS NECESSÁRIOS.....	5
5. PROCEDIMENTOS.....	6
5.1. Pré-requisitos.....	6
5.2. Instaladores .....	9
5.3. Instalação e Configuração do JDK.....	9
5.4. Instalação do EAP .....	10
5.5. Pós-Instalação.....	10
5.6. Criação do usuário de gerência do JBoss.....	14
5.7. Testando a Instalação.....	17
5.8. Outras Tarefas Pós-Instalação.....	19
6. CONSIDERAÇÕES FINAIS.....	20
7. DOCUMENTOS DE REFERÊNCIA.....	20
8. ANEXO I - Instalação do Apache com Mod_Cluster.....	21

## **1. OBJETIVO**

Documentar a instalação do servidor de aplicação Red Hat JBoss EAP 7, para que esta tecnologia esteja padronizada em todos os data centers onde estiver disponibilizada.

## **2. CONTEXTUALIZAÇÃO**

A instalação do Red Hat JBoss EAP já é realizada pelas equipes de suporte dos data centers (RJ, SP e DF) da DATAPREV. Porém, devido a nova versão (EAP 7) e em atendimento às necessidades das equipes de desenvolvimento, foi necessária a elaboração deste documento. Dessa forma, espera-se padronizar a instalação desta tecnologia para que a mesma siga os padrões e sua instalação seja idêntica em todos os data centers.

## **3. VISÃO GERAL**

Este documento descreve em detalhes a instalação e configuração de um ambiente JBoss e tem o objetivo de servir como base para a instalação e configuração de novos ambientes desta tecnologia.

OBS:

Este documento não substitui os manuais oficiais de instalação da Red Hat.

## **4. CONHECIMENTOS NECESSÁRIOS**

- Conceitos e comandos básicos de Linux;
- Conceitos e comando básicos de rede TCP/IP;
- Conceitos de Java SE e Java EE;
- Conceitos de servidor de aplicação Java;
- Conceitos de servidores HTTP.

Recomenda-se a leitura dos documentos abaixo:

- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/7.0/introduction-to-jboss-eap/introduction-to-jboss-eap>
- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/7.0/getting-started-guide/getting-started-guide>
- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/7.0/installation-guide/installation-guide>

## 5. PROCEDIMENTOS

### 5.1. Pré-requisitos

#### 5.1.1. Verificar se o servidor permite acesso

É necessário verificar se foi finalizada a instalação básica do servidor, compreendida pelo sistema operacional RHEL7 ou superior, definições de rede, firewall, armazenamento, liberação de acesso SSH aos grupos de suporte e permissão para SUDO.

O arquivo `/etc/sudoers` deve incluir o registro abaixo para permitir trocas de usuário com as credenciais LDAP do administrador do servidor de aplicação.

```
%supapl ALL = (root) PASSWD: /bin/su
```

#### 5.1.2. Verificar a versão do Sistema Operacional

Verificar qual o SO instalado. Apesar do JBoss executar em quase todas as plataformas existentes no mercado, a fim de termos o suporte "FULL" da Red Hat, convém que o Sistema Operacional seja o Red Hat Enterprise Linux 7.1 ou versão superior (7.1+).

Caso o SO instalado seja Red Hat, vários comandos estão disponíveis para confirmar a versão do SO instalado, como por exemplo: `cat /etc/redhat-release`.

#### 5.1.3. Usuário do Sistema Operacional

Verificar a existência do usuário "jboss", conforme abaixo:  
`uid=59815(jboss) gid=59010(jboss) groups=59010(jboss)`

#### 5.1.4. Limites do sistema operacional

Configurar o arquivo `"limits.conf"`, adicionando os valores abaixo para o usuário "jboss":

jboss	soft	nofile	65536
jboss	hard	nofile	63536
jboss	soft	nproc	16384
jboss	hard	nproc	16384
jboss	soft	memlock	unlimited
jboss	hard	memlock	unlimited
jboss	soft	stack	2048
jboss	hard	stack	2048

#### 5.1.5. Sistema de Arquivos

O servidor deverá ser provisionado com uma partição `/u01` com um tamanho mínimo de 50 GB exclusiva para a instalação do servidor de aplicação. Caso o servidor não possua essa partição, retornar a máquina para que a equipe de plataforma provisione da forma correta.

A vantagem dessa partição dedicada ao servidor de aplicação, serve para que o servidor, físico ou virtual, continue funcionando caso a partição `/u01` fique sem espaço livre e seja possível fazer intervenção para recuperar o serviço.

Para este tipo de servidor recomendamos a utilização das seguintes configurações:

Mapeamento	Tamanho	Montagem
<code>/dev/mapper/vg_local-lv_root</code>	10 GB	<code>/</code>
<code>tmpfs</code>	8 GB	<code>/dev/shm</code>
<code>/dev/sda1</code>	291 MB	<code>/boot</code>
<code>/dev/mapper/vg_local-lv_home</code>	5 GB	<code>/home</code>
<code>/dev/mapper/vg_local-lv_u01</code>	100 GB	<code>/u01</code>
<code>/dev/mapper/vg_local-lv_var</code>	15 GB	<code>/var</code>

#### 5.1.4. Arquivo de HOSTS

Verificar se o arquivo `/etc/hosts` contém as informações necessárias. O servidor tem de ter seu hostname apontando para o endereço IP da interface de dados e também a entrada referente a interface de loopback.

Remover ou comentar qualquer entrada referente a IPV6.

Exemplo:

```
# Endereço IP      hostname.domínio  hostname #
#-----#
127.0.0.1          localhost         localhost.localdomain
10.0.122.121       v151p999         v151p999.pprevnet
```

**ATENÇÃO:** Caso as informações não existam, devem ser inseridas.

### 5.1.5. Verificar o Endereço IP

Através do comando "ifconfig" confirmar o endereço IP do interface de dados do servidor, pois ele será necessário para configurar o JBoss em sua instalação (geralmente eth0). É importante também que o IP seja o mesmo declarado no arquivo /etc/hosts.

Exemplo:

```
eth0      Link encap:Ethernet  HWaddr 00:50:56:9F:18:0D  
          inet addr:10.0.177.85  Bcast:10.0.177.255  Mask:255.255.255.0  
          inet6 addr: fe80::250:56ff:fe9f:180d/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:13096220 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:13031597 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:3656574130 (3.4 GiB) TX bytes:16236648914 (15.1 GiB)
```

### 5.1.6. Definições de Ambiente para o usuário "jboss"

No arquivo **.bash\_profile** do usuário "jboss", serão necessárias algumas configurações para a instalação do JBoss EAP 7.

```
export JBOSS_HOME=/u01/app/jboss/eap  
export JAVA_HOME=/u01/app/jdk  
export TEMP=/tmp  
export PATH=$JBOSS_HOME/bin:$JAVA_HOME/bin:$JAVA_HOME/jre/bin:$PATH
```

**OBS:** o caminho (path) do "JBOSS\_HOME" também aponta para um link simbólico, da mesma forma que o JAVA.

Obviamente, o link deverá ser criado (utilizando o usuário "root") para que aponte para o diretório de instalação do JBoss. Lembrando que o "x" deve ser substituído pelo número correto da versão instalada.

```
ln -s /u01/app/jboss/jboss-eap-7.0/u01/app/jboss/eap
```



## 5.2. Instaladores

Para criação dos ambientes de desenvolvimento, homologação e produção devem ser utilizados os pacotes de software disponíveis no site da Red Hat e Oracle. Antes de iniciar a instalação, confirme qual versão será utilizada e faça o download para o servidor à ser instalado.

<b>SOFTWARE</b>	<b>INSTALADOR</b>	<b>LINK</b>
Red Hat JBoss EAP 7	jboss-eap-7.0.0.zip	<a href="http://developers.redhat.com/products/eap/download/">http://developers.redhat.com/products/eap/download/</a>
JAVA SE - JDK 8 (pré-requisito)	jdk-8u92-linux-x64.tar.gz	<a href="http://www.oracle.com/technet/work/java/javase/downloads/index.html">http://www.oracle.com/technet/work/java/javase/downloads/index.html</a>

## 5.3. Instalação e Configuração do JDK

A instalação deve ser executada utilizando o usuário "jboss" e todos os arquivos devem pertencer a esse usuário.

- Criar o diretório "/u01/app/oracle/jdk/";
- Copiar o instalador para o diretório "/u01/app/oracle/jdk/";
- Acessar o diretório "/u01/app/oracle/jdk/";
- Extrair os arquivos do instalador (jdk-8u92-linux-x64.tar.gz);
- Será criado um diretório (jdk1.8.0\_92). Mover esse diretório para "jdk\_current":

```
mv jdk1.8.0_92 jdk-current
```

- Efetuar "LOGOUT" e "LOGIN" novamente;
- Verificar a versão do java:

```
$> java -version
java version "1.8.0_92"
Java(TM) SE Runtime Environment (build 1.8.0_92)
Java HotSpot(TM) 64-Bit Server VM (build 25.92, mixed mode)
```

## 5.4. Instalação do EAP

### 5.4.1. Criação dos diretórios

O destino do instalador (por padrão) deve ser /u01/app/jboss/. Portanto, temos que criar o diretório:

```
mkdir -p /u01/app/jboss
```

Certifique-se que o dono e grupo sejam "jboss".

```
chown jboss:jboss /u01/app/jboss
```

### 5.4.2. Disponibilizando o Jboss

Agora devemos copiar o arquivo do instalador para esse diretório. O arquivo, geralmente, está em formato ".zip". Todos arquivos a partir de "/u01/app/jboss" devem ser do jboss (owner e group jboss).

```
cp jboss-eap-7.0.0.zip /u01/app/jboss/  
cd /u01/app/jboss/  
unzip jboss-eap-7.0.0.zip
```

## 5.5. Pós-Instalação

Após disponibilizar o JBoss no servidor devemos configurar o modo operacional do serviço, podendo ser domain ou standalone. Na maioria dos casos o JBoss será configurado para trabalhar em modo domain.

**OBS:** Na verdade, não há diferença de funcionalidades entre estes modos operacionais. A grande diferença é a maneira de administrar:

- No modo "**domain**", os servidores podem ser gerenciados centralizadamente, a partir do MASTER;
- No modo "**standalone**", a configuração é independente para cada servidor e, obviamente, este modo operacional faz sentido apenas para servidores que não farão parte de clusters ou farms, como por exemplo, servidores de desenvolvimento e testes.

### 5.5.1. Modo STANDALONE

A configuração do modo "standalone" é bastante simples, bastando para isso referenciar o IP do servidor no devido arquivo de configuração.

Exemplo:

```
cd /u01/app/jboss/jboss-eap-7.0/standalone/configuration  
vim standalone.xml
```

Nesse arquivo, dentro da TAG <interfaces>, deveremos configurar o IP do servidor em todas as interfaces (management, public e unsecure) conforme abaixo, alterando o endereço da interface de loopback (127.0.0.1) para o IP da interface de dados do servidor (obtido no passo 5.1.5 deste documento):

```
<interfaces>  
  <interface name="management">  
    <inet-address value="$  
{jboss.bind.address.management:127.0.0.1}"/>  
  </interface>  
  <interface name="public">  
    <inet-address value="{jboss.bind.address:127.0.0.1}"/>  
  </interface>  
</interfaces>
```

### 5.5.2 Modo DOMAIN

Em modo "Domain" temos um servidor MASTER e um ou mais SLAVES. Para efeito didático, neste item vamos configurar 1 Master e 1 Slave. Caso exista mais de um slave, cada um deve ter nome único: slave1, slave2, etc.

#### 5.5.2.1 Instalação do MASTER

Para configurar o MASTER, deveremos alterar o arquivo "host.xml" no servidor master. Antes de fazermos as alterações, recomendamos fazer um backup de segurança desse arquivo.

Exemplo:

```
vim /u01/app/jboss/jboss-eap-7.0/domain/configuration/host.xml
```

1) No início do arquivo, a TAG <host name> deve estar configurada como "master":

```
<host name="master" xmlns="urn:jboss:domain:1.3">
```

2) A TAG <domain-controller> deve estar configurada como "local":

```
<domain-controller>  
  <local/>  
</domain-controller>
```

3) Na TAG <interfaces> deveremos alterar de 127.0.0.1 para o IP da interface de dados do servidor:

```
<interfaces>  
  <interface name="management">  
    <inet-address value="$  
{jboss.bind.address.management:127.0.0.1}"/>  
  </interface>  
  <interface name="public">  
    <inet-address value="{jboss.bind.address:127.0.0.1}"/>  
  </interface>  
</interfaces>
```

4) O "host.xml" padrão vem configurado com três instâncias de servidores disponíveis. Como estaremos utilizando a máquina apenas para administração dos nós do cluster, essas instâncias devem ser comentadas ou removidas:

```
<servers>
  <server name="server one" group="main server group">
  </server>
  <server name="server two" group="main server group" auto-
start="true">
    <socket bindings port-offset="150"/>
  </server>
  <server name="server three" group="other server group" auto-
start="false">
    <socket bindings port-offset="250"/>
  </server>
</servers>
```

### 5.5.2.2 Instalação do SLAVE

Esta instalação é feita no servidor slave, alterando o mesmo arquivo com os dados necessários. Portanto, não esquecer de fazer o backup de segurança do arquivo `"/u01/app/jboss/jboss-eap-7.0/domain/configuration/host.xml"`

Exemplo:

```
vim /u01/app/jboss/jboss-eap-7.0/domain/configuration/host.xml
```

1) No início do arquivo, a TAG `<host name>` deve estar configurada como "slave1", "slave2", "slave3", etc:

```
<host name="slave1" xmlns="urn:jboss:domain:1.3">
```

2) A TAG `<domain-controller>` deve estar configurada com o IP do "MASTER" onde irá conectar:

```
<domain-controller>
  <remote security-realm="ManagementRealm">
    <discovery-options>
      <static-discovery name="primary" protocol="$
{jboss.domain.master.protocol:remote}" host="$
{jboss.domain.master.address:x.x.x.x}" port="$
{jboss.domain.master.port:9999}"/>
    </discovery-options>
  </remote>
</domain-controller>
```

3) Na TAG <interfaces> deveremos alterar de 127.0.0.1 para o IP da interface de dados do servidor slave:

```
<interfaces>
  <interface name="management">
    <inet-address value="{jboss.bind.address.management:127.0.0.1}"/>
  </interface>
  <interface name="public">
    <inet-address value="{jboss.bind.address:127.0.0.1}"/>
  </interface>
</interfaces>
```

4) O "host.xml" padrão vem configurado com três instâncias de servidores disponíveis. Como estaremos utilizando uma instância por slave, devemos remover as outras duas instâncias:

```
<servers>
  <server name="server-one" group="main-server-group">
    </server>
<server name="server-two" group="main-server-group" auto-
start="true">
  <socket-bindings port-offset="150"/>
</server>
<server name="server-three" group="other-server-group" auto-
start="false">
  <socket-bindings port-offset="250"/>
</server>
</servers>
```

## 5.6. Criação do usuário de gerência do JBoss

Em ambos os modos será necessário criar usuários de gerência para o correto funcionamento do jboss. No modo "standalone", considerando que temos apenas 1 servidor, basta criar um usuário; No modo "domain", precisamos criar o usuário de gerência e um usuário de aplicação para cada "slave", que será usado pelo slave para se conectar ao master.

Para a criação de usuários, o JBoss fornece o script "add-user.sh". Portanto, execute este script em cada servidor do cluster.

```
cd /u01/app/jboss/jboss-eap-7.0/bin/
./add-user.sh
```

Qual o tipo de usuário você deseja adicionar?

- a) Usuário de Gerenciamento (mgmt-users.properties)
- b) Usuário do Aplicativo (application-users.properties)

(a):

Insira os detalhes do novo usuário.

Realm (ManagementRealm) :

Nome do usuário : dsaaadmin

Senha : \*\*\*\*\*

About to add user 'dsaaadmin' for realm 'ManagementRealm'

Isto está correto? sim/não? sim

O 'dsaaadmin' do usuário adicionado ao '/u01/app/jboss/jboss-eap-6.1/standalone/configuration/mgmt-users.properties' do arquivo

O 'dsaaadmin' do usuário adicionado ao '/u01/app/jboss/jboss-eap-6.1/domain/configuration/mgmt-users.properties' do arquivo

Este novo usuário será usado para o processo AS ser conectado a outro processo AS?

ex.: para um controlador host slave conectando a outro mestre ou para uma conexão Remota de servidor para chamadas EJB do servidor

sim/não? sim

Adicione o seguinte à definição server-identities do <secret value="ZGF0YXByZXZAMjAxMw==" /> para representar o usuário

```
<server-identities>  
  <secret value="ZGF0YXByZXZAMjAxMw==" />  
</server-identities>
```

Para o usuário que conecta o slave ao master:

```
./add-user.sh
```

Qual o tipo de usuário você deseja adicionar?

- a) Usuário de Gerenciamento (mgmt-users.properties)
- b) Usuário do Aplicativo (application-users.properties)

(a):

Insira os detalhes do novo usuário.

Realm (ManagementRealm) :

Nome do usuário : slave1

Senha : \*\*\*\*\*

Insira novamente a Senha :

About to add user 'slave1' for realm 'ManagementRealm'

Isto está correto? sim/não? sim

O 'slave1' do usuário adicionado ao '/u01/app/jboss/jboss-eap-6.1/standalone/configuration/mgmt-users.properties' do arquivo

O 'slave1' do usuário adicionado ao '/u01/app/jboss/jboss-eap-6.1/domain/configuration/mgmt-users.properties' do arquivo

Este novo usuário será usado para o processo AS ser conectado a outro processo AS?

ex.: para um controlador host slave conectando a outro mestre ou para uma conexão Remota de servidor para chamadas EJB do servidor

sim/não? sim

Adicione o seguinte à definição server-identities do <secret value="ZGF0YXByZXZAMjAxMw==" /> para representar o usuário

**Dica:** Criar a senha do usuário de gerência igual aos usuários dos slaves, dessa forma teremos apenas uma TAG <server-identities> no arquivo host.xml. Essa senha deve conter obrigatoriamente caracteres alfanuméricos e pelo menos 1 caracter especial.

Detalhe: host.xml

```
<security-realm name="ManagementRealm">
  <server-identities>
    <secret value="ZGF0YXByZXZAMjAxMw==" />
  </server-identities>
  <authentication>
    <local default-user="$local"/>
    <properties path="mgmt-users.properties" relative-to="jboss.domain.config.dir"/>
  </authentication>
</security-realm>
```



## 5.7. Testando a Instalação

Esta etapa testa se o servidor está iniciando sem erros. Nenhum teste mais específico será executado. Precisamos saber apenas se o JBoss iniciou e se estamos com acesso a interface gráfica de gerência.

### 5.7.1 Testando o modo STANDALONE

Para o teste do modo "standalone" deveremos executar o script: standalone.sh

```
"u01/app/jboss/jboss-eap-7.0/bin/standalone.sh":  
=====
```

JBoss Bootstrap Environment

JBOSS\_HOME: /u01/app/jboss/jboss-eap-7.0

JAVA: /u01/app/jdk/bin/java

JAVA\_OPTS: -server -verbose:gc -Xloggc:"/u01/app/jboss/jboss-eap-7.0/standalone/log/gc.log" -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=5 -XX:GCLogFileSize=3M -XX:-TraceClassUnloading -Xms1303m -Xmx1303m -XX:MetaspaceSize=96M -XX:MaxMetaspaceSize=256m -Djava.net.preferIPv4Stack=true -Djboss.modules.system.pkgs=org.jboss.byteman -Djava.awt.headless=true

```
=====
```

Se nada estiver errado a tela acima será exibida durante a inicialização. A mensagem abaixo será exibida no final do LOG:

```
10:27:16,301 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: JBoss  
EAP 7.0.0.GA (WildFly Core 2.1.2.Final-redhat-1) iniciado em 10398ms -  
Iniciado 607 de 899 serviços (os serviços 415 são passivos ou em demanda)
```

### 5.7.2 Testando o modo DOMAIN

Para o teste do modo "domain" deveremos executar o script domain.sh primeiro no master e, somente depois que estiver no ar, executamos nos "slaves":

1) No master, executar:

```
u01/app/jboss/jboss-eap-7.0/bin/domain.sh
```

```
[Host Controller] 10:16:04,578 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: JBoss EAP 7.0.0.GA (WildFly Core 2.1.2.Final-redhat-1) iniciado em 3196ms - Iniciado 51 de serviços 52 (os serviços 15 são passivos ou em demanda)
```

Se tudo estiver OK, a mensagem acima será vista no final do LOG.

2) Nos Slaves, executar:

```
u01/app/jboss/jboss-eap-7.0/bin/domain.sh
```

```
[Server:server-one] 12:59:25,716 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: JBoss EAP 7.0.0.GA (WildFly Core 2.1.2.Final-redhat-1) iniciado em 5480ms - Iniciado 296 de serviços 582 (os serviços 383 são passivos ou em demanda)
```

Se tudo estiver OK, a mensagem acima será vista no final do log do host slave. Poderemos verificar também nos logs do host master, o registro do slave.

```
[Host Controller] 13:21:35,601 INFO [org.jboss.as.domain] (slave-request-threads - 1) WFLYSRV0025: Host slave remoto registrado "slave1", JBoss EAP 7.0.0.GA (WildFly Core 2.1.2.Final-redhat-1).
```

## **5.8. Outras Tarefas Pós-Instalação**

### **5.8.1 Cadastro dos Procedimentos na Ferramenta ATIV**

Após finalizar a instalação do servidor e efetuar os testes necessários, devemos configurar os procedimentos na ferramenta ATIV de acordo com o ambiente e os containers existentes.

Essa tarefa é imprescindível para manutenção e funcionamento dos sistemas no servidor, possibilitando as intervenções pelas Equipes de Suporte de 1º, 2º e 3º níveis.

### **5.8.2 Cadastro do Ambiente Recém-Instalado no SCRIPT de DEPLOY da DSAA.**

Sem a execução deste passo, ficamos impossibilitados de executar atualizações no ambiente em questão.

Ambientes de Homologação e Produção, obrigatoriamente devem ser cadastrados no Sistema de DEPLOY da DSAA.

### **5.8.3 Solicitação de Cadastro do Ambiente no SDM (Service Desk Manager)**

Sem a execução deste passo, os desenvolvedores ficam impossibilitados de solicitar atualizações no ambiente em questão via SDM.

Devemos enviar e-mail para a equipe do SDM, contendo o mesmo nome do ambiente cadastrado no script de deploy e informar também o Cliente/Inquilino.

## 6. CONSIDERAÇÕES FINAIS

Para que todas as funcionalidades estejam operacionais e que o aproveitamento pleno da solução seja atingido, recomendamos que todas as instruções contidas neste documento e no documento de padronização desta tecnologia sejam respeitadas.

Nenhuma modificação deverá ser realizada nos ambientes da solução sem consulta prévia aos elaboradores do documento.

Esta tecnologia deverá ser configurada de forma uniforme, independentemente do DC (datacenter) que esteja efetuando o fornecimento ou contingenciamento.

## 7. DOCUMENTOS DE REFERÊNCIA

- <https://access.redhat.com/articles/2026253>
- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/>
- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/7.0/configuration-guide/chapter-4-network-and-port-configuration>
- <https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/7.0/paged/installation-guide/chapter-2-preparing-for-installation>
- [http://www-planejamentotic/arquivos/DITF\\_PTI\\_014\\_2015\\_M.pdf](http://www-planejamentotic/arquivos/DITF_PTI_014_2015_M.pdf)

## 8. ANEXO I - Instalação do Apache com Mod\_Cluster

### Introdução

O objetivo deste manual é instalar o mod\_cluster, que é um balanceador de carga semelhante ao mod\_jk, porém seu mapa de configuração é criado de forma dinâmica. Toda a instalação será baseada em pacotes binários, utilizando a versão do WebServer Apache do próprio RedHat.

Instalador	Descrição
RedHat	RHEL 7.1+ x86_64
Apache HTTP	2.4.xx el7 x86_64
Mod Cluster	Final Linux x64

**Nota:** Todo o procedimento será executado como **ROOT**. Para acesso a Internet no servidor, exporte a variável de proxy: **export http\_proxy=http://10.0.151.211:8080**

```
host:# vim .bash_profile
```

Inserir no final do arquivo:

```
export http_proxy=http://10.0.151.211:8080
```

### Instalação do Apache HTTP

O processo de instalação será via YUM, utilizando o repositório próprio do RedHat:

```
host:# yum install httpd httpd-tools mod_ssl
```

Criando estrutura centralizada:

Utilizaremos a partição apartada da raiz do SO "u01", nela centralizaremos a configuração e módulos personalizados, logs e conteúdo estático do servidor.

Crie a seguinte estrutura:

```
host:# mkdir -p /u01/app/apache  
host:# cd /u01/app/apache
```

```
host:# mkdir cgi-bin conf htdocs logs modules
host:# ln -sd /var/www/error
host:# ln -sd /var/www/icons
```

Algo semelhante a isso será montado:

```
host:/u01/app/apache # ls -la
total 20
drwxr-xr-x 2 root root 4096 Sep 20 14:31 cgi-bin
drwxr-xr-x 2 root root 4096 Sep 23 08:18 conf
lrwxrwxrwx 1 root root 15 Sep 20 14:42 error -> /var/www/error/
drwxr-xr-x 2 root root 4096 Sep 20 14:04 htdocs
lrwxrwxrwx 1 root root 15 Sep 20 14:42 icons -> /var/www/icons/
drwxr-xr-x 2 root root 4096 Sep 23 07:13 logs
drwxr-xr-x 2 root root 4096 Sep 20 14:15 modules
```

Acesse /u01/app/apache/conf, vamos mover as configurações padrão do apache para o novo diretório:

```
host:# mv /etc/httpd/conf/httpd.conf
/u01/app/apache/conf/httpd.conf
host:# mv /etc/httpd/conf.d/ssl.conf
/u01/app/apache/conf/mod_ssl.conf
```

Vamos agora criar os links nos locais originais:

```
host:# ln -s /u01/app/apache/conf/httpd.conf
/etc/httpd/conf/httpd.conf
host:# ln -s /u01/app/apache/conf/mod_ssl.conf
/etc/httpd/conf.d/mod_ssl.conf
```

### Escolhendo Módulo de Multiprocessamento:

O modo padrão da distribuição do Apache HTTP é o MPM Prefork, que prove um processamento sem Threads, sendo útil para processos pesados como é o caso do PHP. Com o mod\_jk, e não é diferente com o mod\_cluster, o servidor torna-se mais performático utilizando o MPM Worker ao Prefork, que é um híbrido envolvendo multi-thread por processo, sendo um único processo pai (parent) é responsável lançar os processos filhos (child), até então como no prefork, porém que cada processo child lançam vários Threads, que ouvem as conexões.

Para fazer a mudança basta trocar os binários em:

```
host:# cd /usr/sbin
host:# mv httpd httpd.prefork
host:# ln -s httpd.worker httpd
```

Ficará assim:

```
host:/usr/sbin # ls -la httpd*
lrwxrwxrwx 1 root root      12 Sep  5 10:58 httpd -> httpd.worker
-rwxr-xr-x 1 root root 368168 Aug  2 09:03 httpd.event
-rwxr-xr-x 1 root root 354816 Aug  2 09:03 httpd.prefork
-rwxr-xr-x 1 root root 367240 Aug  2 09:03 httpd.worker
```

### Configuração Básica:

#### **httpd.conf**

```
# Apache Default
Timeout 60
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 5

# Configurações MPM Worker
<IfModule worker.c>
ThreadLimit      150
StartServers     3
MaxClients       2400
MinSpareThreads  5
MaxSpareThreads  20
ThreadsPerChild  150
MaxRequestsPerChild 2000
ServerLimit      16
ThreadStackSize  1024
</IfModule>

# Comente o mod_proxy_balancer pois conflita com mod_cluster
#LoadModule proxy_balancer_module modules/mod_proxy_balancer.so

# Descomente mod_unique_id, é necessário para o mod_security
LoadModule unique_id_module modules/mod_unique_id.so
```

```
# Liga Status estendido - Zabbix
ExtendedStatus On

# Desliga a necessidade de nomes canônicos
UseCanonicalName Off

# Novo DocumentRoot e Remoção do Indexes
DocumentRoot "/u01/app/apache/htdocs"

<Directory "/u01/app/apache/htdocs">
...
    Options FollowSymLinks
...
</Directory>

# Desliga a verificação de DNS
HostnameLookups Off

# Rotação de LOGS
ErrorLog "|rotatelogs /u01/app/apache/logs/error_log.%Y%m%d 86400
-180"

CustomLog "|rotatelogs /u01/app/apache/logs/access_log.%Y%m%d
86400 -180" combined

# Novo caminho CGI
ScriptAlias /cgi-bin/ "/u01/app/apache/cgi-bin/"

<Directory "/u01/app/apache/cgi-bin">
...
</Directory>

# Configura o Server Status e Info para local e rede 10.0
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1 10.0
</Location>

<Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
```



```
Allow from 127.0.0.1 10.0
</Location>
```

### **mod\_ssl.conf**

```
# Rotação dos LOGS

ErrorLog "|rotatelogs /u01/app/apache/logs/error_log.%Y%m%d 86400
-180"

TransferLog "|rotatelogs /u01/app/apache/logs/access_log.%Y%m%d
86400 -180"

CustomLog "|rotatelogs /u01/app/apache/logs/access_log.%Y%m%d
86400 -180" \
    "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
```

### **Instalação do ModCluster**

A instalação consiste em baixar o pacote com as bibliotecas no site [jboss.org](http://jboss.org) e criar um arquivo de configuração para o módulo:

```
host:# cd /tmp
host:# wget http://downloads.jboss.org/mod_cluster/1.2.0.Final/mod_cluster-1.2.0.Final-
linux2-x64-so.tar.gz
host:# cd /u01/app/apache/modules
host:# tar -xvzf /tmp/mod_cluster-1.2.0.Final-linux2-x64-so.tar.gz
host:# chown root:root *
```

Ficará assim:

```
host:/u01/app/apache/modules # ll
total 428
-rwxr-xr-x 1 root root 51211 Sep 20 14:15 mod_advertise.so
-rwxr-xr-x 1 root root 198676 Sep 20 14:14 mod_manager.so
-rwxr-xr-x 1 root root 124787 Sep 20 14:14 mod_proxy_cluster.so
-rwxr-xr-x 1 root root 43143 Sep 20 14:14 mod_cluster_slotmem.so
```

```
host:# cd /usr/lib64/httpd/modules
```

```
host:# ln -s /u01/app/apache/modules/mod_manager.so
host:# ln -s /u01/app/apache/modules/mod_proxy_cluster.so
host:# ln -s /u01/app/apache/modules/mod_cluster_slotmem.so
host:# ln -s /u01/app/apache/modules/mod_advertise.so
```

Crie o seguinte arquivo de configuração em /u01/app/apache/conf:

```
host:# vim /u01/app/apache/conf/mod_cluster.conf

# Requeridos pelo mod_cluster
LoadModule advertise_module modules/mod_advertise.so
LoadModule manager_module modules/mod_manager.so
LoadModule proxy_cluster_module modules/mod_proxy_cluster.so
LoadModule cluster_slotmem_module modules/mod_cluster_slotmem.so

# MOD_CLUSTER_ADDS
# Adjust to you hostname and subnet.
<IfModule manager_module>
  Listen 11968
  <VirtualHost _default_:11968>
    <Directory />
      Order deny,allow
      Deny from all
      Allow from 127.0.0 10.0
    </Directory>
    KeepAliveTimeout 300
    MaxKeepAliveRequests 0
    #ServerAdvertise on http://@IP@:6666
    AdvertiseFrequency 5
    #AdvertiseSecurityKey secret
    #AdvertiseGroup @ADVIP@:23364
    EnableMCPMReceive
    ManagerBalancerName mycluster
    <Location /mod_cluster-manager>
      SetHandler mod_cluster-manager
      Order deny,allow
      Deny from all
      Allow from 127.0.0 10.0
    </Location>
  </VirtualHost>
</IfModule>
```

Agora falta incluir as configurações no conf.d, para isso:

```
host:# ln -s  
/u01/app/apache/conf/mod_cluster.conf  
/etc/httpd/conf.d/mod_cluster.conf
```

### **Instalação do ModSecurity**

A instalação discorrerá através do YUM utilizando o repositório externo EPEL:

#### **1) Verifique a versão do Red Hat:**

```
host:# cat /etc/redhat-release
```

##### **1.a) Para Red Hat 6:**

```
host:# wget http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-  
release-6-8.noarch.rpm
```

```
host:# rpm -ivh epel-release-6-8.noarch.rpm
```

##### **1.b) Para Red Hat 5:**

```
wget http://dl.fedoraproject.org/pub/epel/5/epel-release-5-  
4.noarch.rpm
```

```
host:# rpm -ivh epel-release-5-4.noarch.rpm
```

```
host:# vim /etc/yum.repos.d/epel.repo
```

#### **2) Configurando o repositório:**

```
# Troque mirrorlist por baseurl
```

```
baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch
```

```
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-  
6&arch=$basearch
```

```
host:# yum makecache
```

```
host:# yum install mod_security
```

Remova:

```
host:# rm -rf /etc/httpd/modsecurity.d
```

```
host:# rm -rf /etc/httpd/conf.d/mod_security.conf
```

Crie o seguinte arquivo de configuração em /u01/app/apache/conf :

```
host:~# vim /u01/app/apache/conf/mod_security.conf

LoadModule security2_module modules/mod_security2.so
<IfModule !mod_unique_id.c>
    LoadModule unique_id_module modules/mod_unique_id.so
</IfModule>
<IfModule mod_security2.c>
# Basic configuration options
SecRuleEngine On
SecRequestBodyAccess On
SecResponseBodyAccess Off

# PCRE Tuning
SecPcreMatchLimit 1000
SecPcreMatchLimitRecursion 1000

# Handling of file uploads
# TODO Choose a folder private to Apache.
# SecUploadDir /opt/apache-frontent/tmp/
SecUploadKeepFiles Off
SecUploadFileLimit 10

# Debug log
SecDebugLog /u01/app/apache/logs/modsec_debug.log
SecDebugLogLevel 0

# Serial audit log
SecAuditEngine RelevantOnly
SecAuditLogRelevantStatus ^5
SecAuditLogParts ABIFHZ
SecAuditLogType Serial
SecAuditLog /u01/app/apache/logs/modsec_audit.log

# Maximum request body size we will
# accept for buffering ----> SETADO PARA 20Mb
SecRequestBodyLimit 20971520

# Store up to 128 KB in memory
SecRequestBodyInMemoryLimit 131072

# Buffer response bodies of up to
# 512 KB in length
SecResponseBodyLimit 524288
```

```
# SecRuleEngine DetectionOnly will only process request/response
bodies
# up to the buffering limit defined and will not block the
transaction.
SecRequestBodyLimitAction Reject

# Initiate XML Processor in case of xml content-type
#
# TODO Uncomment this rule if you wish to parse
#      text/xml requests using the XML parser.  Note
#      that this may cause considerable overhead in processing
#      text/xml requests.
#RecRule REQUEST_HEADERS:Content-Type "text/xml" \
#
# "id:'200000',phase:1,t:none,t:lowercase,pass,nolog,ctl:requestBody
Processor=XML"

# Verify that we've correctly processed the request body.
# As a rule of thumb, when failing to process a request body
# you should reject the request (when deployed in blocking mode)
# or log a high-severity alert (when deployed in detection-only
mode).
SecRule REQBODY_ERROR "!@eq 0" \
    "id:'200001', phase:2,t:none,log,deny,status:400,msg:'Failed
to parse request body.',logdata:'%{reqbody_error_msg}',severity:2"

# By default be strict with what we accept in the multipart/form-
data
# request body. If the rule below proves to be too strict for your
# environment consider changing it to detection-only. You are
encouraged
# _not_ to remove it altogether.
SecRule MULTIPART_STRICT_ERROR "!@eq 0" \
    "id:'200002',phase:2,t:none,log,deny,status:44,msg:'Multipart
request body \
    failed strict validation: \
    PE %{REQBODY_PROCESSOR_ERROR}, \
    BQ %{MULTIPART_BOUNDARY_QUOTED}, \
    BW %{MULTIPART_BOUNDARY_WHITESPACE}, \
    DB %{MULTIPART_DATA_BEFORE}, \
    DA %{MULTIPART_DATA_AFTER}, \
    HF %{MULTIPART_HEADER_FOLDING}, \
    LF %{MULTIPART_LF_LINE}, \
    SM %{MULTIPART_MISSING_SEMICOLON}, \
```

```
IQ %{MULTIPART_INVALID_QUOTING}, \
IP %{MULTIPART_INVALID_PART}, \
IH %{MULTIPART_INVALID_HEADER_FOLDING}, \
FL %{MULTIPART_FILE_LIMIT_EXCEEDED}"

# Did we see anything that might be a boundary?
SecRule MULTIPART_UNMATCHED_BOUNDARY "!@eq 0" \
    "id:'200003',phase:2,t:none,log,deny,status:44,msg:'Multipart
parser detected a possible unmatched boundary.'"

# Some internal errors will set flags in TX and we will need to
look for these.
# All of these are prefixed with "MSC_". The following flags
currently exist:
#
# MSC_PCRE_LIMITS_EXCEEDED: PCRE match limits were exceeded.
#
SecRule TX:/^MSC_/ "!@streq 0" \
    "id:'200004',phase:2,t:none,deny,msg:'ModSecurity
internal error flagged: %{MATCHED_VAR_NAME}%"

#
# Defeat HTTP fingerprinting
#

# Change server signature
SecServerSignature "DTP WebServer"

# Add X-Powered-By header to mimic IIS
Header set X-Powered-By "DTP AppServer"

# Remove the ETag header
Header unset ETag
</IfModule>
```

Agora falta incluir as configurações no conf.d, para isso:

```
host:# ln -s
/u01/app/apache/conf/mod_security.conf
/etc/httpd/conf.d/mod_security.conf
```

**Demais Módulos:**

Repita o processo de criação em /u01/app/apache/conf

**mod\_deflate.conf**

```
<IfModule deflate_module>
  SetOutputFilter DEFLATE
  SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|png)$ \
    no-gzip dont-vary
  SetEnvIfNoCase Request_URI \
    \.(?:exe|t?gz|zip|bz2|sit|rar)$ \
    no-gzip dont-vary
  SetEnvIfNoCase Request_URI \.pdf$ no-gzip dont-vary

  BrowserMatch ^Mozilla/4 gzip-only-text/html
  BrowserMatch ^Mozilla/4\.0[678] no-gzip
  BrowserMatch \bMSIE !no-gzip !gzip-only-text/html

  DeflateFilterNote Input instream
  DeflateFilterNote Output outstream
  DeflateFilterNote Ratio ratio

  #LogFormat '"%r" %{outstream}n/%{instream}n (%{ratio}n%)'
  deflate
  #CustomLog logs/deflate_log deflate
</IfModule>
```

Criando o link:

```
host:# ln -s
/u01/app/apache/conf/mod_deflate.conf
/etc/httpd/conf.d/mod_deflate.conf
```