

# Webové aplikace a stránky na platformě Apache + Tomcat

## Apache 1.x (NW 6.0)

Apache je obvykle nainstalován v adresáři `sys: \apache`

### Důležité podadresáře:

`\bin` - spustitelné soubory  
`\conf` - konfigurační soubory  
`\logs` - log soubory apache

### Spouštění:

Obvykle je vytvořen NCF soubor, ve kterém se uvede s jakým konfiguračním souborem se apache spustí, např. `sys: \apache\bin\apache.nlm -f sys: \apache\conf\tdp.conf`  
Pokud je tedy třeba měnit konfiguraci je nutné nejdříve zjistit jakým způsobem se apache spouští a odkud si tedy bere konfigurační soubor, může to být i na jiném svazku.  
Spustí-li se apache bez uvedení konfiguračního souboru, pak je použit soubor `httpd.conf` z adresáře `\conf` v instalaci apache.

### Konfigurace:

#### Soubor `*.conf`

Na začátku konfiguračního souboru jsou obecné parametry instance apache. Za zmínku stojí zejména `KeepAlive`, jeho zapnutím se web obvykle rapidně zrychlí, ale může to způsobit problémy v IE při protokolu `https`. (viz možné problémy)

```
Timeout 300
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
ThreadsPerChild 50
ThreadStackSize 81920
MaxRequestsPerChild 0
```

Zde se specifikuje, které moduly se musí pro běh této instance apache natáhnout. Pokud apache vyžaduje nějaký další modul, tak se při jeho startu objeví chybová hláška a po odklepnutí se apache ukončí.

```
LoadModule lcgimodule modules/mod_lcgi.nlm
LoadModule nds_auth_module modules/mod_nds.nlm
LoadModule tlsmodule modules/mod_tls.nlm
LoadModule headers_module modules/headers.nlm
```

Zde musí být vyjmenovány všechny kombinace portů a adres na kterých tato instance apache poslouchá

```
Port 80
Listen 192.168.0.51:80
<IfModule mod_tls.c>
    SecureListen 192.168.0.51:443 "SSL CertificateDNS"
    SecureListen 192.168.0.51:444 "SSL CertificateDNS"
    SecureListen 192.168.0.51:1443 "SSL CertificateDNS"
```

```
</IfModule>
```

Zde je zkonfigurován modul `jk`, který umožňuje propojení apache s tomcatem přes konektor `ajp13`. Důležité je správně zadat cestu ke konfiguračnímu souboru `nworkers.properties`. Pokud tento soubor neexistuje, pak použijte vzorový soubor z adresáře `tomcat\j k` na svazku `TEMP`.

```
LoadModule jk_module modules/mod_jk.nlm
```

```
JkWorkersFile "SYS:/tomcat/41/conf/jk/nworkers.properties"  
JkLogFile "SYS:/tomcat/41/logs/mod_jk.log"
```

Nejdůležitější částí je konfigurace adresářů webu a propojení na servlety a jsp

Direktiva `DocumentRoot` definuje kořenový adresář kde jsou umístěny jednotlivé weby. Pokud chceme aby nějaký web byl umístěn v jiném kořeni, tak je možné definovat `VirtualHost` nebo `Alias`.

```
DocumentRoot "sys:docs"
```

Direktivy `Directory` a `Location` slouží k specifikaci dalších parametrů daného adresáře nebo subadresáře. Nastavení atributů je vždy platné pro celý podstrom specifikovaný daným adresářem, nicméně může být pro daný podadresář předefinováno. Pokud není v definici atributů nějaký atribut uveden je nastavení převzato z nadřazeného adresáře. Atribut `deny from all` implicitně zakazuje přístup k souborům v daném podstromu ze všech adres, parametr `allow from all` naopak v daném podstromu přístup k souborům povoluje.

```
<Directory "/">  
    Options FollowSymLinks  
    AllowOverride None  
    Order deny,allow  
    deny from all  
</Directory>  
<Directory "sys:docs">  
    Options Indexes FollowSymLinks MultiViews  
    AllowOverride None  
    Order allow,deny  
    allow from all  
</Directory>
```

Specifikace propojení apache s tomcatem je umožněna direktivou `JkMount`. Pro každou webovou aplikaci je nutné namapovat všechny podadresáře s jsp stránkami a servlety, přičemž je třeba brát v úvahu, že adresáře použité v servletech mohou být virtuální (jsou definovány v tagu `servlet-mappings` v souboru `web.xml` dané aplikace). Mapování jsp a servletů je provedeno pomocí wildcardu, přičemž servlety se obvykle mapují pouze pomocí hvězdičky `*` (pokud jsou ve stejném podadresáři i jsp stránky, pak se toto mapování vztahuje i na ně) a jsp stránky pomocí jejich přípony, pokud mají stránky více možných přípon, pak musí být uvedeny všechny. Na konci každého řádku je uvedeno jméno `ajp13` konektoru, které je zavedeno v souboru `nworkers.properties`.

```
JkMount /bois/*.jsp ajp13  
JkMount /bois/jsp/*.jsp ajp13  
JkMount /bois/web/*.jsp ajp13  
JkMount /bois/web/*.jspf ajp13  
JkMount /bois/include/*.jspf ajp13
```

následující subadresáře servletů jsou virtuální a jsou definovány ve `web.xml` dané aplikace

```
JkMount /bois/config/* ajp13  
JkMount /bois/servlets/* ajp13
```

```
JkMount /bois/cewolf/* ajp13
```

Aliasy umožňují webu přistupovat i k souborům, které jsou mimo adresářovou strukturu specifikovanou direktivou `DocumentRoot`.

```
Alias /icons/ "sys:/apache/icons/"
```

Pomocí direktivy `VirtualHost` můžeme určitý web nebo webovou aplikaci zkonfigurovat samostatně. Pro každou kombinaci IP adresy a portu může existovat pouze jeden `VirtualHost`, přičemž pro obsah této direktivy platí stejná pravidla jako pro defaultní konfiguraci adresářů webu. Pokud není uvedena direktiva `DocumentRoot`, pak je kořenový adresář stejný jako defaultní.

```
<VirtualHost 192.168.0.17:82>
```

specifikace kořenového adresáře webu a jména serveru

```
DocumentRoot "web:apache/grandis/"
ServerName grandis.bonpart.cz
```

specifikace dalších atributů webu v tomto adresáři

```
<Location "web:apache/grandis">
    Order allow,deny
    Allow from all
</Location>
```

následuje mapování servletů a jsp

```
JkMount /*.jsp ajp13
JkMount /grandis/*.jsp ajp13
JkMount /grandis/servlet/* ajp13
```

uzavření direktivy

```
</VirtualHost>
```

## Možné problémy:

### 1) Problémy se stahováním souborů přes SSL

Pokud chceme u starších verzí IE 5.0, 5.5 umožnit stahování souborů přes https, pak je nutné downgradovat protokol SSL a vypnout `keepalive`. Obojí je možné udělat, tak aby se to aplikovalo pouze na IE. Toto nastavení bohužel vede ke značnému zpomalení.

```
<VirtualHost _default_:54443>
    BrowserMatch "MSIE" nokeepalive ssl-unclean-shutdown downgrade-1.0 force-
response-1.0
</VirtualHost>
```

```
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown downgrade-1.0
force-response-1.0
```

### 2) Občasné výpadky obrázků, javascriptu nebo stylesheetů v IE

V tomto případě pomáhá (2 až 3-násobné) zvýšení hodnot `MaxKeepAliveRequests` a `KeepAliveTimeout`.

### **3) Apache nenabíhá**

Zkontrolovat zda není chyba v konfiguračním souboru (neznámý prvek, překlep, apod.)

Je rovněž možné, že chybí některý modul použitý v direktivě `LoadModule` v konfiguračním souboru. Nebo naopak chybí tato direktiva pro některý z modulů, který je v konfiguraci použit. Další možností je poškozený nebo neexistující certifikát použitý pro SSL.

## Tomcat 4.0, 4.1, 5.0

Tomcat je obvykle instalován v adresáři `sys: tomcat\VERZE`, tedy např. `sys: tomcat\41`. Při instalaci na clusteru to však může být na jiném svazku.

### Důležité podadresáře:

`\bin` - zde se nachází NCF skript a knihovny pro spuštění tomcat.  
`\logs` - logy tomcata, webových aplikací a obvykle také ajp13 konektoru  
`\conf` - konfigurační soubory – `server.xml`, `web.xml`  
`\webapps` - Zde jsou obvykle nainstalované příslušné weby a webové aplikace. Aplikace je však možné umístit i do jiného adresáře. Je ovšem třeba modifikovat konfiguraci kontextu této aplikace nebo celého kontejneru. (viz. popis struktury `server.xml`)  
`\temp` - cache JAR souborů.  
`\work` - rozbalené soubory webových aplikací a zkompileované JSP stránky

### Spouštění:

Tomcat se obvykle spouští NCF skriptem `tomcat41.ncf` nebo `tomcat40.ncf` podle verze tomcatu. Tento skript není součástí instalace tomcatu a pokud tedy provádíte novou instalaci tak bude nutné ho vytvořit nebo upravit skript z jiné instalace. Na začátku skriptu by neměla chybět řádka s natažením javy, pokud je tomcat první spuštěná javovská aplikace je nutné před použitím příkazu `envset` spustit javu.

Příklad spouštěcího skriptu (jméno skriptu např. `tomcat41.ncf`):

```
load java -server
```

```
envset C41_HOME=SYS:\tomcat\41
envset C41_BASE=%C41_HOME
envset C41_TMPDIR=%C41_HOME\temp
envset C41_ENDORSED=%C41_HOME\bin;%C41_HOME\common\endorsed\
envset C41_CPATH=%JAVA_HOME\lib\tools.jar
envset C41_CPATH=%C41_CPATH;%C41_HOME\common\endorsed\xercesImpl.jar
envset C41_CPATH=%C41_CPATH;%C41_HOME\common\endorsed\xmlParserAPIs.jar
envset C41_CPATH=%C41_CPATH;%C41_HOME\bin\bootstrap.jar;%C41_HOME\common\lib
```

```
java -Xms64m -Xmx256m -envCWD=%C41_HOME -Djava.endorsed.dirs=%C41_ENDORSED
-classpath %C41_CPATH -Dcatalina.base=%C41_BASE -Dcatalina.home=%C41_HOME
-Djava.io.tmpdir=%C41_TMPDIR org.apache.catalina.startup.Bootstrap %1 %2 %3
```

Spuštění javy v režimu server by mělo být součástí `autoexec.ncf` před spuštěním první javovské aplikace.

Tomcat se pomocí tohoto skriptu nainstaluje příkazem `tomcat41 start` a ukončí příkazem `tomcat41 stop`. Předpokládám zde, že cesta k tomuto skriptu je rovněž součástí `autoexec.ncf`.

Pokud se tomcat neukončí standardním způsobem, pak je nutné ho ukončit příkazem:

```
java -kill<ID>
```

kde `<ID>` je číslo javovského procesu tomcata, které zjistíme příkazem `java -show`.

### Instalace a konfigurace:

Instalace tomcatu spočívá ve zkopírování adresářové struktury buď přímo z instalačního archívu nebo z jiné instalace. Pokud použijete jinou instalaci, pak je nutné vyčistit pomocné adresáře a odstranit původní webové aplikace. Konkrétně by tedy měl být vymazán obsah adresářů `\temp`, `\work` a `\logs`. V adresáři `\webapps` by pak měly zůstat pouze tyto soubory a adresáře:

adresáře ROOT a webdav (u tomcat 4.0 i adresáře admin a manager)  
soubory admin.xml a manager.xml

### *Struktura souboru \conf\server.xml*

Tento soubor obsahuje celkovou konfiguraci tomcatu a často také konfiguraci jednotlivých webových aplikací. (příklad je vzat z tomcatu verze 4.1)

V tagu Server je uveden port na kterém je možné poslat signál k jeho ukončení.

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

zde je uvedena konfigurace uživatelů aplikace manager pro spravu tomcatu

```
<GlobalNamingResources>
  <Resource name="UserDatabase" auth="Container"
            type="org.apache.catalina.UserDatabase"
            description="User database that can be updated and saved">
  </Resource>
  <ResourceParams name="UserDatabase">
    <parameter>
      <name>factory</name>
      <value>org.apache.catalina.users.MemoryUserDatabaseFactory</value>
    </parameter>
    <parameter>
      <name>pathname</name>
      <value>conf/tomcat-users.xml</value>
    </parameter>
  </ResourceParams>
</GlobalNamingResources>
<Service name="Tomcat-Standalone">
```

uvnitř tagu Service následují definice konektorů, zde je uveden příklad konektoru typu ajp13 pro spojení přes SSL v tomcatu verze 4.1. Je zde rovněž uveden port pro tento konektor, stejný musí být uveden v souboru nworkers.properties.

```
<Connector className="org.apache ajp.tomcat4.Ajp13Connector"
           port="9009" minProcessors="5" maxProcessors="75"
           acceptCount="10" debug="0"/>
<Engine name="Standalone" defaultHost="localhost" debug="0">
```

Následující modul dumpuje veškeré requesty na tomcat do logu, není tedy vyloženě potřebný pro běh tomcatu

```
<Valve className="org.apache.catalina.valves.RequestDumperValve"/>
```

Dále je zde obvykle definice logu pro daný Service

```
<Logger className="org.apache.catalina.logger.FileLogger"
        prefix="catalina." suffix=".log"
        timestamp="true"/>
```

Zde je definice způsobu autentikace do aplikace manager a eventuelně i do dalších aplikací, které využívají tomcat k autentizaci

```
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
        debug="0" resourceName="UserDatabase"/>
```

Tag `Host` definuje kontejner pro webové aplikace, je rovněž možné pomocí atributu `appBase` změnit adresář, ve kterém jsou webové aplikace umístěny.

```
<Host name="localhost" debug="0"
  className="org.apache.catalina.core.StandardHost" appBase="webapps"
  autoDeploy="true"
  configClass="org.apache.catalina.startup.ContextConfig"
  contextClass="org.apache.catalina.core.StandardContext"
  deployXML="true"
  errorReportValveClass="org.apache.catalina.valves.ErrorReportValve"
  mapperClass="org.apache.catalina.core.StandardHostMapper"
  unpackWARs="true" liveDeploy="true">

<Logger className="org.apache.catalina.logger.FileLogger"
  directory="logs" prefix="localhost." suffix=".log"
  timestamp="true"/>
```

Zde mohou být uvedeny kontexty jednotlivých aplikací a kontext `ROOT`u, kde je obvykle manažovací aplikace `tomcatu` (`ROOT` je rovněž využit v `groupwise webaccess`)

```
<Context path="" docBase="ROOT" debug="0"/>

</Host>

</Engine>

</Service>

</Server>
```

### *Soubor `\conf\jk\jwworkers.properties`*

zde je uveden adresář kde je nainstalován tomcat, který poskytuje implementaci konektoru `ajp13`

```
workers.tomcat_home=sys:/tomcat/41
```

zde je uveden seznam jmen konektorů obvykle se používají varianty `ajp13`

```
worker.list=ajp13,ajp13_gw
```

následuje konfigurace jednotlivých konektorů, přičemž je vždy použit název daného konektoru, důležitý je zejména port na kterém konektor poslouchá a který je rovněž uveden v konfiguraci konektoru v `server.xml`.

```
worker.ajp13.port=9009
worker.ajp13.host=127.0.0.1
worker.ajp13.type=ajp13
```

```
worker.ajp13_gw.port=9019
worker.ajp13_gw.host=127.0.0.1
worker.ajp13_gw.type=ajp13
```

### *Soubor `\conf\web.xml`*

V tomto souboru je definováno implicitní nastavení servletů a jsp. Za zmínku stojí zejména parametry `jsp`, kde je možné nastavit tag `pooling` a kódování ve kterém jsou kompilovány jsp stránky.

```

<servlet>
  <servlet-name>jsp</servlet-name>
  <servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class>
  <init-param>
    <param-name>logVerbosityLevel</param-name>
    <param-value>WARNING</param-value>
  </init-param>
  <init-param>
    <param-name>fork</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param>
    <param-name>enablePooling</param-name>
    <param-value>>true</param-value>
  </init-param>
  <init-param>
    <param-name>javaEncoding</param-name>
    <param-value>Cp1250</param-value>
  </init-param>
  <load-on-startup>3</load-on-startup>
</servlet>

```

Rovněž se může stát, že jsou v nějaké aplikaci použity jsp stránky s jinou příponou než jsp, v takovém případě je nutné přidat mapování pro tyto stránky pod mapování přípony jsp.

```

<servlet-mapping>
  <servlet-name>jsp</servlet-name>
  <url-pattern>*.jsp</url-pattern>
</servlet-mapping>

```

## Instalace a konfigurace webových aplikací

### **Kontext aplikace:**

Kontext aplikace je zkonfigurován buď přímo v `server.xml` nebo také tzv. xml fragmentem, což je soubor jehož název je stejný jako název kontextu aplikace a který je umístěn v adresáři `\webapps` (`\webapps\vykazy.xml`). Pokud tento fragment neexistuje podívá se tomcat do `server.xml` a pokud není kontext uvede ani tam, použije se pro kontext defaultní nastavení, jméno kontextu je v takovém případě určeno jménem adresáře ve `\webapps` nebo jménem webového archívu (`vykazy.war`, `vykazy.ear`).

```

<Context path="/vykazy" docBase="vykazy" debug="3"
  reloadable="true" crossContext="false"
  charsetMapperClass="cz.charset.CharsetMapper">
  <Logger className="org.apache.catalina.logger.FileLogger"
    prefix="vykazy-" suffix=".log"
    timestamp="true"/>
</Context>

```

### **Definice webu (deskriptor webu):**

Deskriptor webu je umístěn v adresáři dané aplikace v podadresáři `WEB-INF` a jmenuje se vždy `web.xml` (`\webapps\vykazy\WEB-INF\web.xml`). V tomto souboru jsou uvedeny parametry aplikace, jsou zde definovány servlety a jejich mapování. Dále jsou zde uvedeny použité tagliby a jejich umístění. Pokud to není nutné, pak by tento soubor neměl být měněn, protože se jedná o součást dané aplikace (soubor může být navíc součástí webového archívu aplikace). V některých případech, kdy se například v tomto souboru vyskytují absolutní cesty nebo nesprávné cesty k taglibům.

Samotná instalace aplikace spočívá v nakopírování adresáře s aplikací do podadresáře `\webapps`



nebo jiného adresáře, který je definován v tagu `Host` v `server.xml` nebo v definici kontextu aplikace. Totéž se vztahuje na webové archívy, s tím rozdílem, že aplikace je místo v podadresáři umístěna uvnitř archívu.

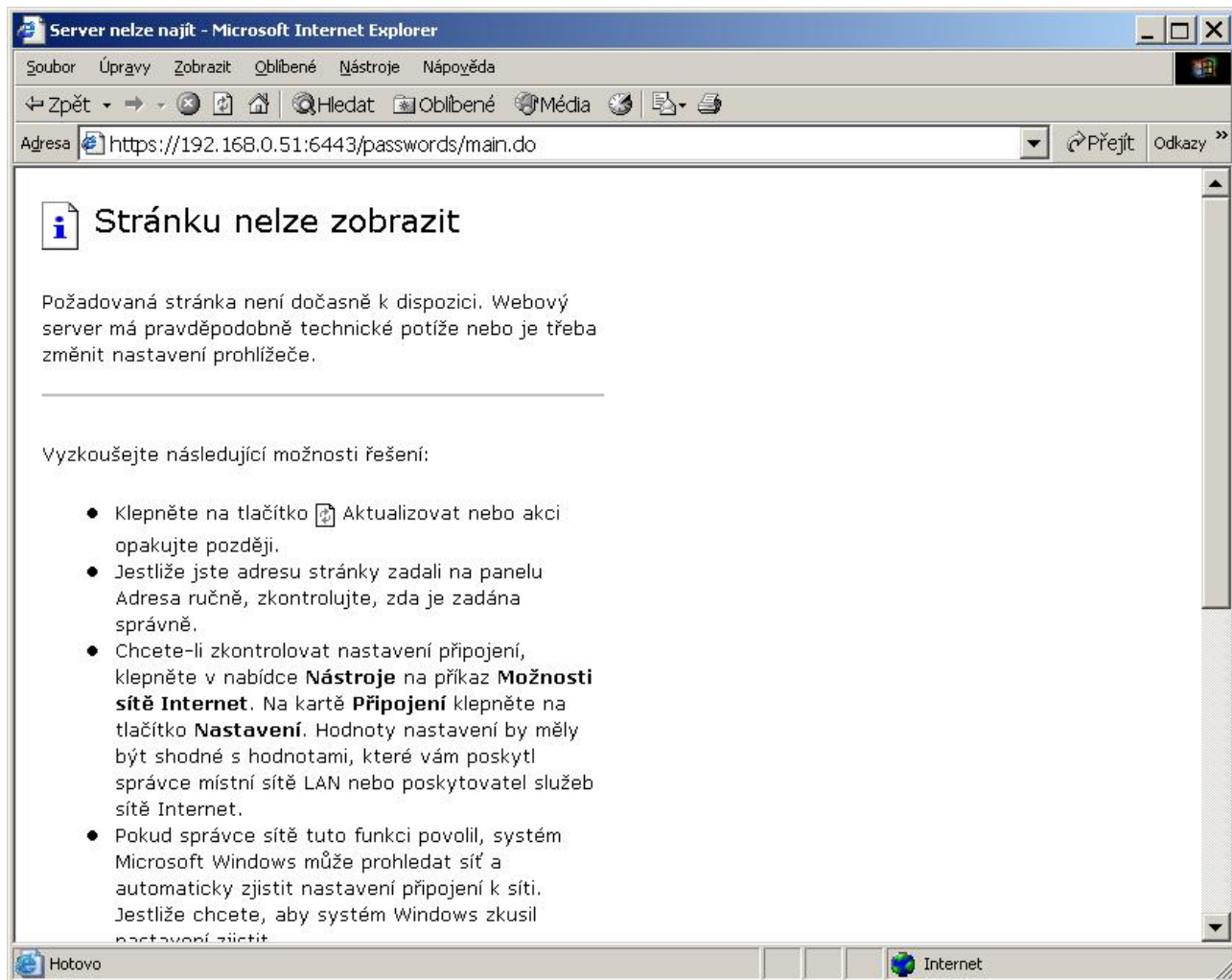
## Možné problémy:

### 1) aplikace není přístupná

V mozilla se zobrazí hláška:



a v IE se zobrazí stránka:



možné příčiny:

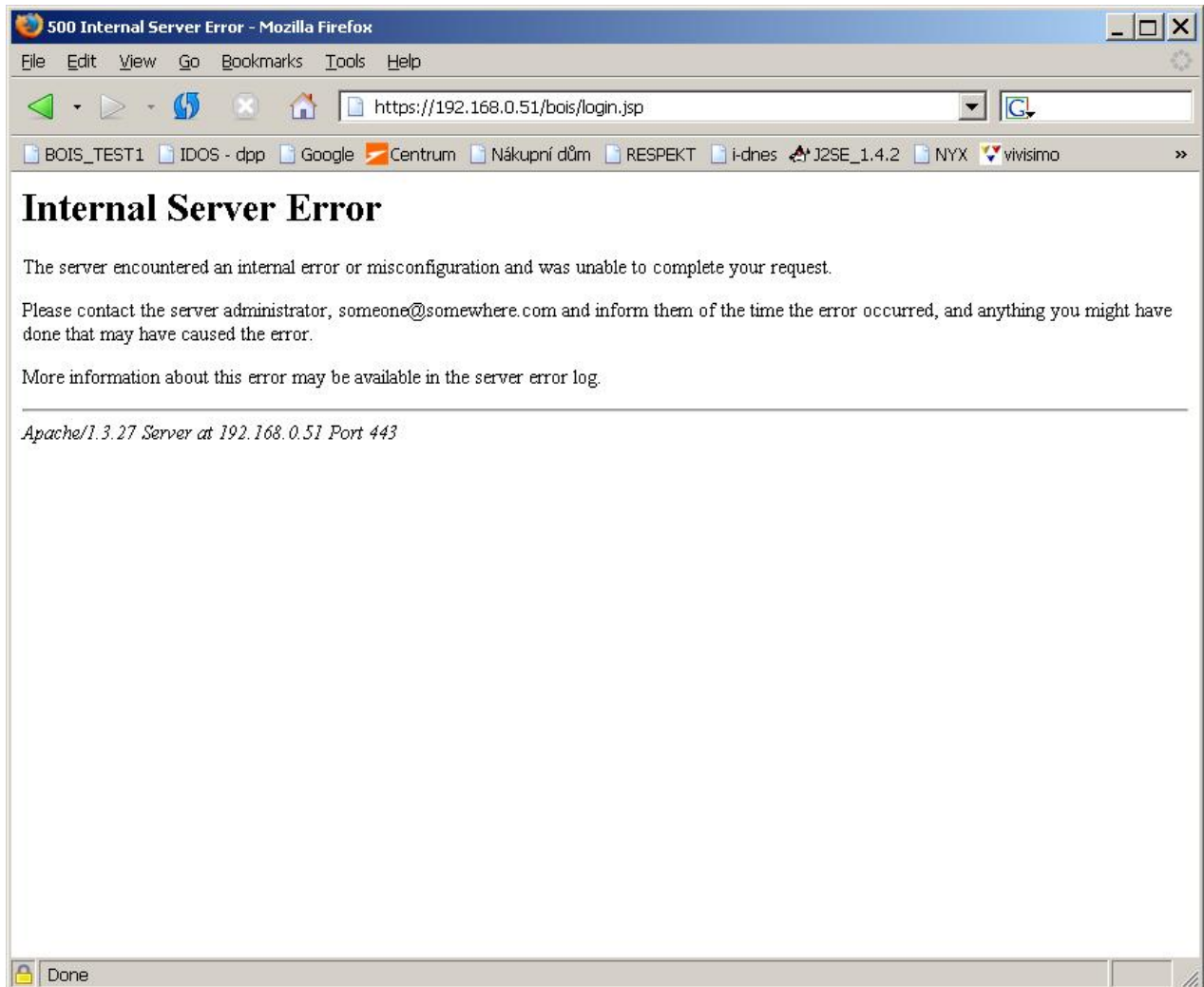
- i) neběží Apache
- ii) daná adresa není přístupná zvenku

řešení:

- zjistit proč neběží apache a spustit ho a pokud běží, tak zkontrolovat nastavení adres
- pokud se spouští více instancí apache, zkontrolovat jejich spouštěcí skripty
- zkontrolovat zda se všechny instance apache spouští v autoexec.ncf
- zkontrolovat zda aplikace funguje na vnitřní adrese, pokud ano, tak zkontrolovat konfiguraci vnější adresy na proxy serveru, borderu, apod.
- zkontrolovat danou adresu pomocí utility TPCON

## 2) aplikace není přístupná

V prohlížeči se zobrazí stránka (stránka se může lišit v různých verzích apache, může být lokalizovaná a může být také modifikována webmasterem, nicméně by měla obsahovat údaje o daném webservru):



možné příčiny:

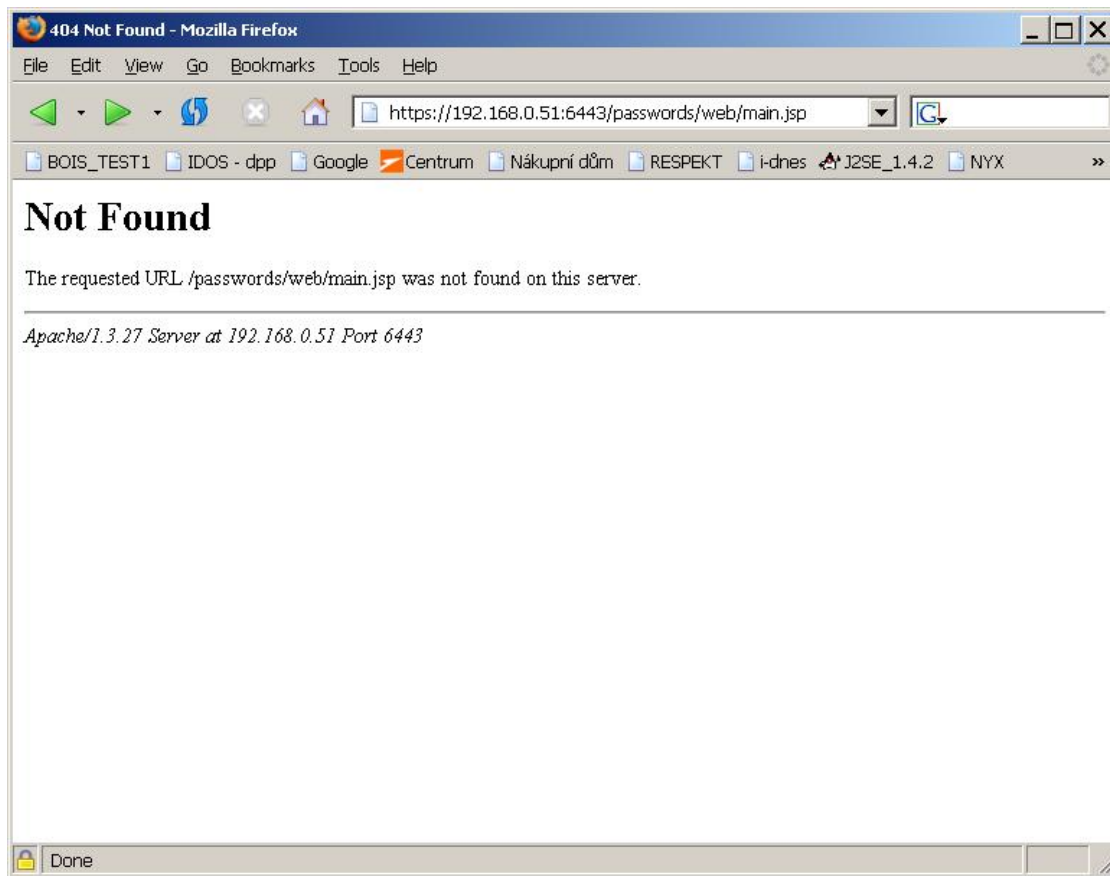
- i) neběží tomcat
- ii) není správně zkonfigurováno propojení apache s tomcatem

řešení:

- pokud neběží tomcat, tak ho spustit (viz. spouštění tomcatu) a případně řešit příčiny proč nenaběhl,
- zkontrolovat zda je spouštění tomcatu v autoexec.ncf,
- zkontrolovat cestu k souboru `nworkers.properties` v konfiguraci apache,
- dále zkontrolovat zda odpovídají porty konektoru v `nworkers.properties` a v definici konektoru `ajp13` v `server.xml`,
- zkontrolovat zda odpovídá název konektoru definovaný v `nworkers.properties` a použitý pro namapování servletů a jsp v konfiguraci apache.

### 3) aplikace nebo její část není přístupná

V prohlížeči se zobrazí stránka (i tato stránka se může mírně lišit v různých verzích apache):



možné příčiny:

- i) v konfiguračním souboru apache není správně definováno mapování pro jsp nebo servlet
- ii) zadané url není správné – může jít například o změnu relativní cesty k dané stránce přesunem aplikace do jiného adresáře

řešení:

- zkontolovat a případně přidat do konfigurace apache příslušné mapování (viz. popis konfigurace apache)
- pokud byla aplikace přesunována z jiného adresáře nebo z jiné verze tomcat, pak je nutné zkontrolovat, zde je správně zkonfigurován kontext aplikace (viz. konfigurace kontextu)
- je rovněž možné, že je uvedeno nesprávné přesměrování na úvodní stránku ve statické části webu (soubor `index.html` v `DocumentRoot` definovaném v konfiguraci apache)

#### 4) aplikace nebo její část není přístupná

V prohlížeči se zobrazí stránka:



možné příčiny:

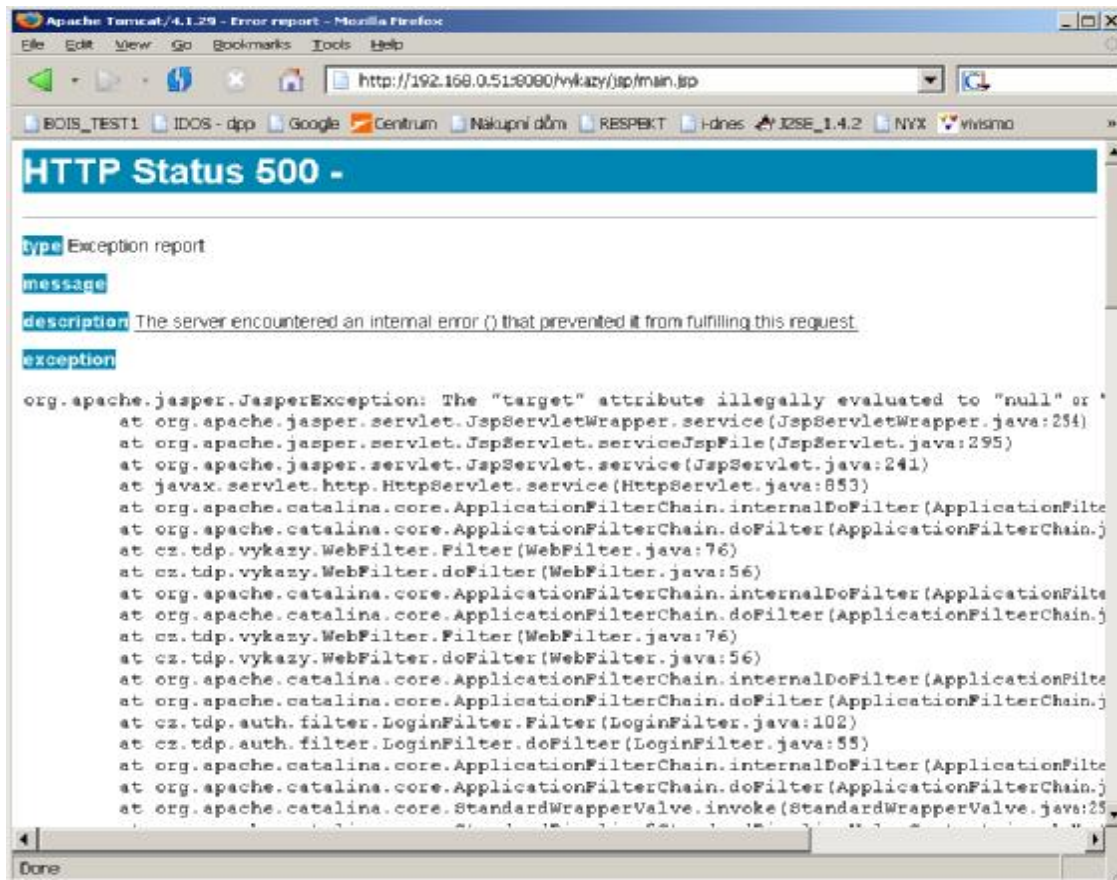
- i) Neběží kontext dané aplikace
- ii) Není správně definováno mapování servletů v definici aplikace (web.xml)

řešení:

- nejdříve je třeba zjistit, zda kontext běží. to lze zjistit tak, že nahlédneme do logu příslušné webové aplikace (např. vykazy. yyyy-MM-dd .log) nebo do logu kontejneru localhost (obvykle localhost\_1.log. yyyy-MM-dd.txt). Pokud kontext nenaběhne, tak o tom bude v jednom z těchto logů zapsána chybová hláška.

## 5) při přístupu na stránku aplikace nastane chyba

V prohlížeči se zobrazí stránka:



možné příčiny:

- i) na stránce aplikace je syntaktická chyba nebo nastala běhová chyba ve stránce či servletu
- ii) chybí knihovna nezbytná pro běh aplikace

řešení:

- nejdříve je třeba vyzkoušet vyčištění cache tomcatu (vymazání obsahu adresářů `\temp` a `\work`).
- poté znovu nastartovat tomcat (viz. spouštění tomcatu)
- v případě, že existuje instalační archív nebo postup, je možné zkusit aplikaci reinstalovat
- pokud nic z toho nepomůže, pak se jedná o chybu aplikace a je třeba kontaktovat autora.

## 6) tomcat nenastartuje

možné příčiny:

- i) chyba v souboru `server.xml` (viz. problematika editace konf. souborů na konci textu)
- ii) chyba ve spouštěcím skriptu tomcata – chybějící knihovna nebo chybí `load java` na začátku.

## 7) porušení kódování češtiny při submitu formuláře

možné příčiny:

- i) V kontextu aplikace musí být specifikován atribut `charsetMapperClass` a nastaven na hodnotu `cz.charset.CharsetMapper` pro kódování `windows-1250` a `cz.charset.CharsetMapperUTF8` pro kódování `UTF-8`.
- ii) V `server.xml` je zapnut `RequestDumper`, který slouží k zápisu informací o POST nebo GET http requestu do logu tomcata. To způsobí, že jsou parametry requestu překódovány do ISO8859-1.
- iii) U tomcat 5.x je nutné u konektoru AJP13 v `server.xml` nastavit atribut `URIEncoding` na požadované kódování GET requestu. Nebo paramter `useBodyEncodingForURI` na `true`, čímž bude použito kódování stránky z níž byl request odeslán.

## **Přesun aplikací z tomcat 4.0 na 4.1**

Verze 4.1 je oproti 4.0 zoptimalizovaná s ohledem na rychlost. Při přesunu aplikace je potřeba si dát pozor zejména na soubory `web.xml`, verze 4.1 má přísnější kontrolu dependencí a tak v něm nemohou být uvedeny odkazy na soubory (obvykle tagliby), které neexistují. Rovněž v souboru `server.xml` jsou změny,

Ve verzi 4.1 je defaultně zapnutý tzv. tag `pooling`, který umožňuje rychlejší načítání jsp stránek.

Některé starší aplikace však `pooling` nepodporují, a pokud si tedy nejste jisti je vhodné tag `pooling` vypnout ve `web.xml` v adresáři `\conf` tomcata (viz. popis `web.xml`).

Dále je nutné aby byla knihovna `tools.jar` z adresáře `sys:java\lib` zkopírována rovněž do adresáře tomcatu `\commons\lib`.

## **Přesun aplikací z tomcat 4.1 na 5.0**

Verze 5.0 je oproti 4.1 ještě více zoptimalizovaná a u aplikací se to viditelně projeví. Pokud aplikace používá standard taglib (soubory `jsp.taglib.standard.jar` v adresáři `WEB-INF\lib`), je nutné je z tohoto adresáře odstranit, protože verze 5.0 vyžaduje verze těchto knihoven pro tuto verzi tomcatu. Tato novější verze se však obvykle nachází v instalaci tomcat 5.0 v adresáři `\shared\lib`, takže je přístupná pro všechny aplikace. Pokud instalujete tomcat 5.0 nanovo, pak je nutné použít záplatu knihovny `\server\lib\jasper-compiler.jar` (správná verze je v adresáři na svazku `TEMP` v adresáři `tomcat\50\server\lib`).

## **Spouštění více instancí Tomcatu**

Pro současný běh více tomcatů stejné verze není nutné spouštět více instancí apache, ale je nutné správně zkonfigurovat porty pro ukončení tomcatů a pro konektory `ajp13` (`nworkers.properties` a `server.xml`).

Pokud se jedná o různé verze tomcatu, pak je nutné spustit více instancí apache a pro každou instanci apache a tomcatu mít zvláštní soubor `nworkers.properties`.

Aby bylo možné rozlišit jednotlivé instance tomcatu je možné upravit spouštěcí skript každé instance tomcatu, takto:

volání třídy `Bootstrap`: `org.apache.catalina.startup.Bootstrap`

nahradíme voláním archívu `bootstrap.jar`, tedy: `-jar sys:tomcat\41\bin\bootstrap.jar`  
uvedená cesta k archívu musí být napsána bez použití proměnných (např. `$C41_HOME`).

Tímto způsobem tedy můžeme při použití příkazu `java -show` rozlišit různé instalace tomcatu.

## **Spouštění tomcat a apache na clusteru**

### **Apache na clusteru**

Při spouštění apache na clusteru je často vhodné mít konfiguraci apache na jiném svazku než na `sys`, takže je pak možné sdílet tuto konfiguraci mezi všemi serveru clusteru. Postup je následující:

1. vytvoří se adresáře `apache\conf` a `apache\logs` na zamýšleném svazku
2. přesune se (nebo vytvoří) konfigurační soubor apache do tohoto adresáře
3. upraví se absolutní cesty k logům a eventuelně k `mime.types` v tomto souboru
4. dále je vhodné upravit `DocumentRoot` a `Directory`, tak aby weby jednotlivých aplikací v této instanci apache mohli být také na jiném svazku (obvykle na stejném jako konfigurace apache a



instalace tomcat)

5. vytvoří nebo změní se load a unload skript příslušného serveru na clusteru

Příklad změny konfigurace (převzato z ČMKBK):

```
PidFile boisvol:apache/logs/httpd.pid
ScoreBoardFile boisvol:apache/logs/apache_status
ErrorLog boisvol:apache/logs/error_log
TypesConfig boisvol:apache/conf/mime.types
```

Příklad nastartování a ukončení apache na clusteru (vzato z load a unload skriptu)

```
start: load address space=apb sys:apache\apache -f boisvol:apache\conf\tdp.conf
```

```
ukončení: unload address space=apb apache
```

### **Tomcat na clusteru**

Pokud chceme spouštět tomcat na clusteru je vhodné mít vyhrazenou jednu či více instalací tomcatu jen pro aplikace. Je ovšem třeba dbát na všechna pravidla pro spouštění více instancí tomcatu (viz. předchozí kapitola). Při instalaci tomcatu na clusteru je třeba postupovat následujícím způsobem:

1. vytvoří se příslušný adresář na zamýšleném svazku
2. přesune se do něj instalace tomcatu spolu se zamýšlenými aplikacemi
3. upraví se cesty ve spouštěcím skriptu tomcata v adresáři `\bin`
4. upraví se cesty v souboru `nworkers.properties`
5. upraví se cesta k `nworkers.properties` v konfiguraci apache
6. zkontroluje se zda se nevyskytují absolutní cesty v `server.xml` a v context deskriptorech jednotlivých aplikací v adresáři `\webapps`
7. upraví nebo zkontrolují se porty pro ajp13 a shutdown tomcatu v `server.xml` a `nworkers.properties`
8. zkontroluje se zda se nevyskytují absolutní cesty v souborech `web.xml` jednotlivých aplikací
9. pokud aplikace využívají kryptografické knihovny, pak je nutné aby na všech serverech clusteru se v instalačním adresáři javy (`sys:java\lib\security`) v souboru `java.security` byly uvedeni všichni provideři (`parametr security.provider.n`)
10. vytvoří nebo změní se load a unload skript příslušného serveru na clusteru

### **Na co si dát pozor při editaci konfiguračních souborů**

Při editaci xml souborů (ty s příponou xml) je třeba si dát pozor na syntaxi tagů. V zásadě jsou v nich dva typy tagů: s tělem `<tagName attr="bl"> .... </tagName>` a bez těla `<tagname attr="blabla"/>`. Komentáře jsou v xml započaty znaky `<!--` a ukončeny znaky `-->`, uvnitř komentáře se může vyskytovat prakticky cokoliv, kromě skupiny znaků, která jej ukončuje. Nejčastějšími chybami jsou:

- i) neuzavření tagu s tělem
- ii) uzavření počátečního tagu jakoby byl bez těla (`<tagName attr="bl"/> .... </tagName>`)

iii) nesprávné ukončení tagu bez těla (vynechání lomítka / nebo i většítka > )

iv) přehlédnutí komentáře – celý tag může být uzavřen v komentáři a je při zpracování ignorován.

První tři chyby ve většině případů vedou k odmítnutí celého konfiguračního souboru. Nebo v případě ii k ignorování těla tagu. Pokud tedy nenabíhá například tomcat, může to být způsobeno právě jednou z chyb i, ii a iii v souboru `server.xml`. Totéž platí pro nenaběhnutí kontextu a soubor `web.xml`

Dalším možným problémem je nedodržení vyšší struktury xml souboru. Což jsou pravidla pro to jaký tag může být vnořen v nějakém jiném tagu, kolikrát se může opakovat, nebo také jaké musí být pořadí tagů. Zejména pořadí tagů je důležité v souboru `web.xml` a jeho nedodržení je častou příčinou nenaběhnutí kontextu aplikace.

V konfiguračním souboru apache se rovněž vyskytují tagy, takže se na něj do určité míry rovněž vztahují pravidla pro editaci tagů v xml.