

Správa serverů a počítačových sítí

Přednáška 5

2020/2021

(ver. 2021-03-30-01)



WWW, servery, skriptovací jazyky

- 1. WWW
 - Oč jde, historie
 - Prostředky HTML, HTTP
- 2. WWW servery
 - Základní vlastnosti, požadavky
 - Klasické použití a chyby
 - Příklady implementace
- 3. Programovací jazyky pro WWW
 - PHP
 - ASP.NET
 - Java
 - Perl, Python, C, C++



1. WWW

- World Wide Web
 - Autor Tim Berners-Lee
 - První WWW koncem roku 1990
- Spolu s maily nejpoužívanější služba internetu
- Hypertextové odkazy, decentralizovanost
- Komunikace typu klient server
- Využívá HTML a HTTP případně rozšíření HTTPS



1.1 WWW – URL, klienti

- URL - Uniform Resource Locator
- Adresa už dnes nemusí být jedinečná – proxy, clustery
- Klientem je webový prohlížeč
 - Internet Explorer, Edge
 - Netscape, Mozilla, Firefox, **Chrome**
 - Safari
 - Links, wget, curl, ...
- Významný rozmach s příchodem vyhledávačů
 - Yahoo, AltaVista, Google, Seznam, Atlas, Pinknet



1.2 WWW - HTML

- HyperText Markup Language
- Značkovací jazyk
- Server jej pouze předává či tvoří, klient jej interpretuje
- V současné verzi 5.2
- Na počátku pouze statický jazyk, následně rozšiřován o další možnosti jako CSS, JavaScript, Applety atd



1.2 WWW – HTML - příklad

```
<!DOCTYPE html>  
<html>  
<head>  
<title>Page Title</title>  
</head>  
<body>  
  
<h1>My First Heading</h1>  
<p>My first paragraph.</p>  
  
</body>  
</html>
```

Zdroj: https://www.w3schools.com/html/html_intro.asp



1.3 WWW - XHTML

- Extensible HyperText Markup Language
- Původně zamýšlen jako následovník HTML, nyní vyvíjen samostatně
- HTML rozšířen o vlastnosti XML
 - Všechny značky ukončené
 - Rozlišuje malá a velká písmena
 - Dokument musí začínat XML deklarací



1.3 WWW – XHTML - příklad

XHTML Example

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
  lang="en">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
      charset=utf-8" />
    <title>XHTML 1.0 Example</title>
  </head>
  <body>
    <p>This is a example.</p>
  </body>
</html>
```

Zdroj: <https://www.slideshare.net/pohjus/introduction-to-xml-xhtml-and-css-presentation>



1.4 WWW - HTTP

- Hypertext Transfer Protocol
- Původně navržen jen pro HTML, nyní díky MIME umí přenášet jakýkoliv soubor
- Běžně používá TCP port 80
- V základu je to protokol je bezstavový
- Následně rozšířen o HTTP cookie
- Základní příkazy : HEAD, GET, POST, PUT, DELETE , CONNECT, TRACE, OPTION



1.4 WWW – HTTP - příkazy

HTTP Methods

GET

.....> Request for a web page or an object from server

PUT

.....> For sending a document to the server

POST

.....> For sending data or information about client to the server

DELETE

.....> Request to Delete an object on the server

HEAD

.....> Request for information about a web page or a document

TRACE

.....> Used to trace the proxies and tunnels in the path from client to server

OPTION

.....> Used to determine server's capabilities

Zdroj: <http://computernetworkingsimplified.in/application-layer/basic-overview-http-protocol/>

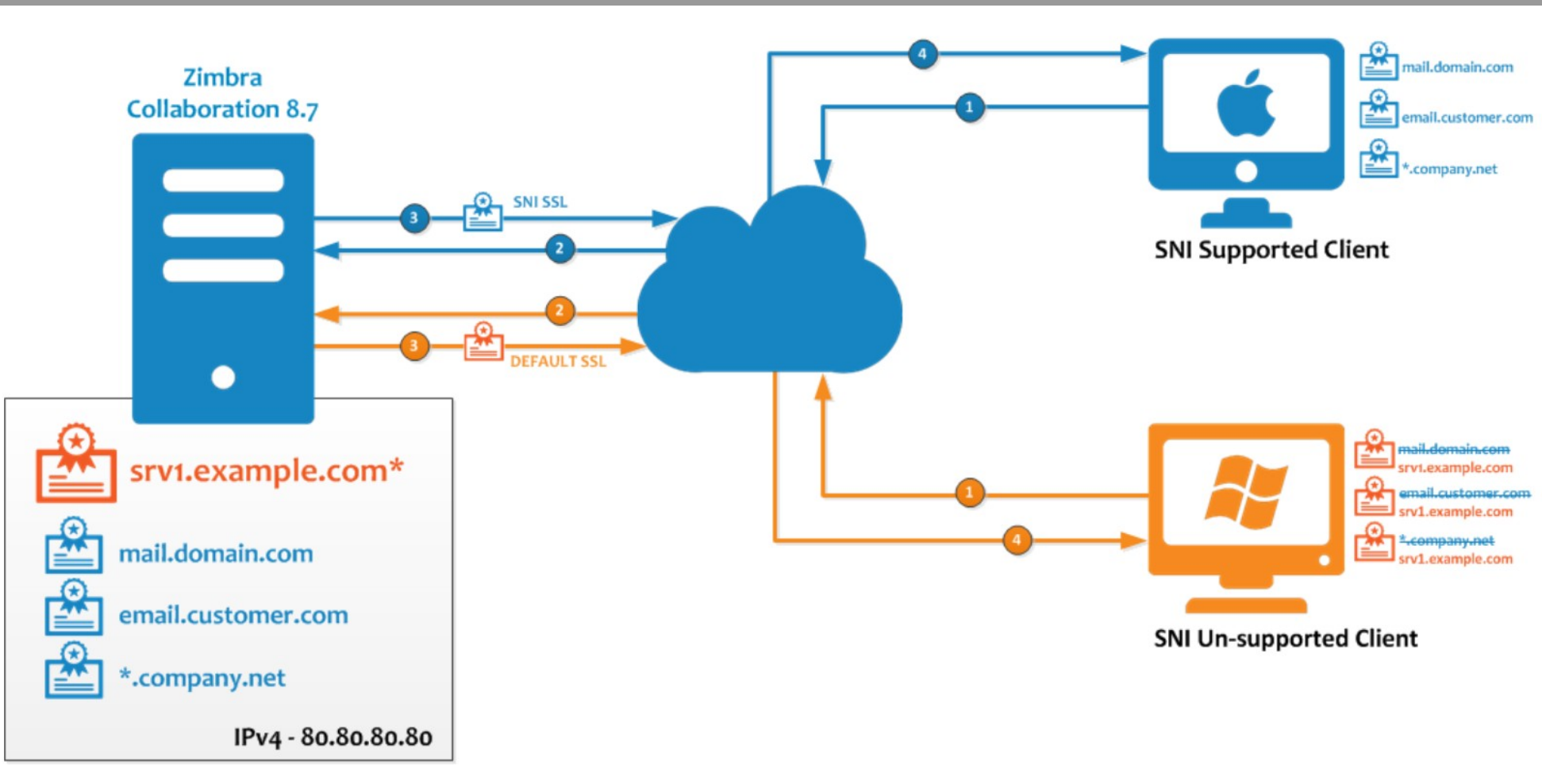


1.5 WWW - HTTPS

- Zabezpečená verze http protokolu
- Klasicky se provozuje na TCP 443
- Nejprve je navázáno šifrované spojení a následně jsou tímto tunelem posílány příkazy a data
- Vlastní IP nebo SNI
- HTTPS server je svázán s certifikátem
 - Vlastní - openssl, podepsané či nepodepsané
 - Placený - Thawte, VeriSign, Digicert
 - Free - Let's Encrypt



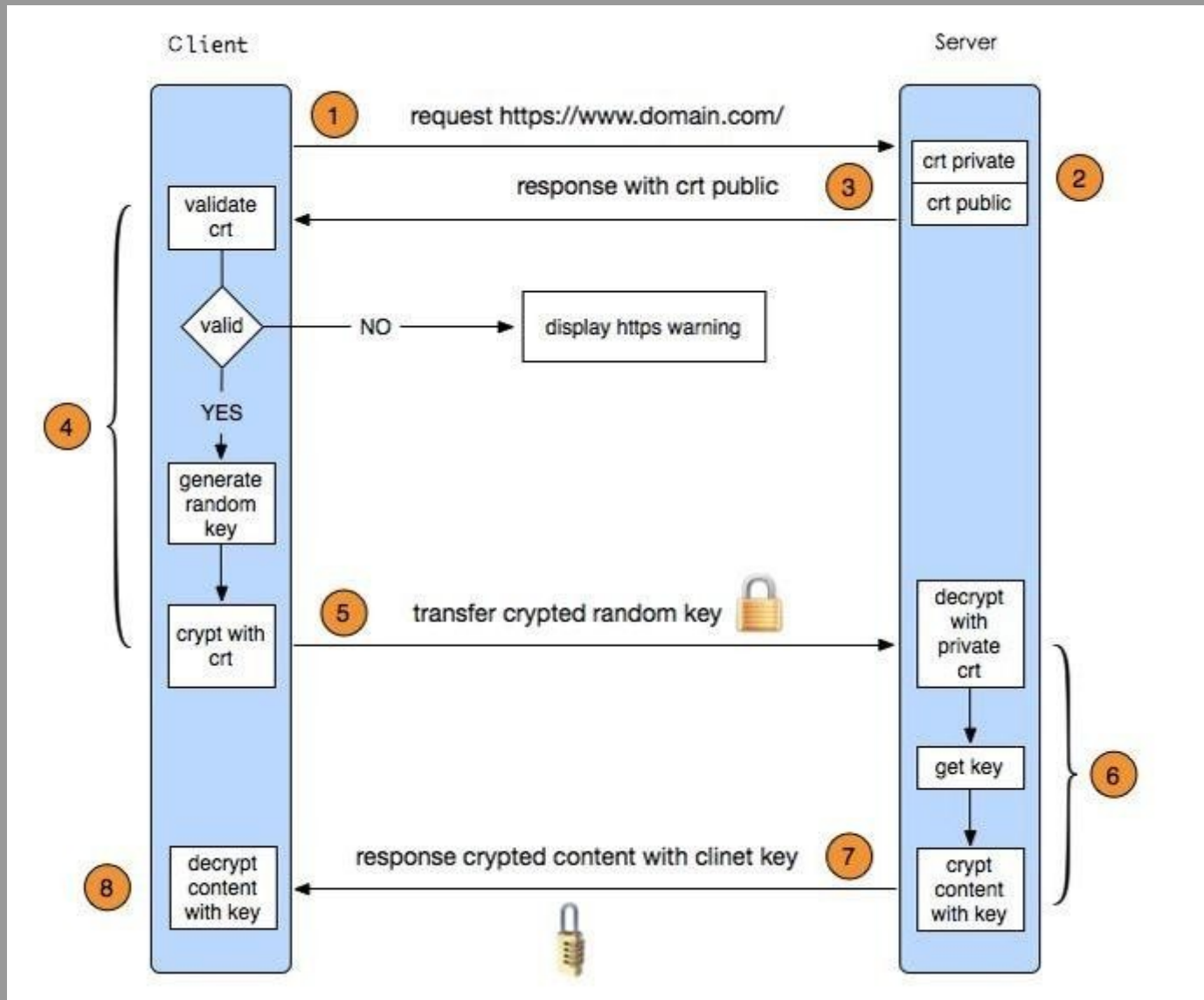
1.5 WWW – HTTPS – příklad SNI



Zdroj: <https://www.zimbra.com/email-server-software/ssl-sni/>



1.5 WWW – HTTPS – příklad SSL



Zdroj: <https://www.programmersought.com/article/94714648612/>



2. WWW servery

- Server poskytuje obsah klientům prostřednictvím HTTP
- Komunikace po TCP port 80 a 443
- Stejný obsah dnes poskytuje více serverů
- Servery jsou dostupné pro všechny běžné platformy
- Dnes mají řadu dostupných modulů pro rozšíření nad rámec prosté distribuce HTML



2.1 WWW servery - konfigurace

- Každý server může obsluhovat velké množství virtuálních serverů
- Je možné kombinovat HTTP a HTTPS na úrovni jednoho serveru
- Základní informace pro virtual jsou doména a uložště
- Doména musí být správně nastavená před startem www-serveru



2.2 WWW servery - konfigurace

- *Keep alive* – podržení spojení pro daného klienta
- *Min a max počet serverů* – třeba nastavovat rozumně hlavně s ohledem na velikost dostupné paměti stroje
- *Timeout* – jak dlouho se čeká, než klient odpoví – pozor na DoS či DDoS
- *User, group* – server by neměl být spuštěn s právy privilegovaného uživatele



2.3 WWW servery - implementace

- Apache, NginX, Lighttpd
 - Unix, MS Windows
- IIS
 - MS Windows
- Boa, Thttpd
 - Unix
- Tomcat, Jetty, Winstone
 - Cokoliv kde je Java



2.4 WWW servery - Apache

- Robustní WWW server
- Prošel už dlouhým vývojem a testováním
- Patrně nejpoužívanější WWW server
- Nyní dostupné verze 1.3, 2.0, 2.2, 2.4
- Základní konfigurace v textových souborech
- Možnost přenést konfiguraci například od MySQL
- Vhodný na menší i rozsáhlé projekty
- Nevhodný na rychle odbavování statického obsahu



2.4 WWW servery - Apache

- Možnost nastavení formátu logovacích souborů
- Modulární systém
- K dispozici je velké množství modulů např:
 - CGI
 - SSL
 - Rewrite
 - PHP5
 - Perl, Python
 - Status a Info
 - Proxy



WWW server – Apache – ukázka konfuračního souboru

```
<VirtualHost *:80>  
    ServerName students.kiv.zcu.cz  
    ServerAlias students.fav.zcu.cz  
    ServerAlias ares.fav.zcu.cz  
    ServerAdmin webmaster@mail.kiv.zcu.cz  
    DocumentRoot /home/www/apache/students.kiv.zcu.cz  
    ErrorLog /var/log/apache2/students.kiv.zcu.cz-error.log  
    CustomLog /var/log/apache2/students.kiv.zcu.cz-access.log  
    Alias /vs/ /home/validator/vs2/data/html/  
    Alias /moodle/ /home/extern/pooh/moodle/  
</VirtualHost>
```



WWW server – Apache – ukázka konfuračního souboru SSL

```
<VirtualHost 147.228.63.10:443>  
    DocumentRoot "/home/www/apache/students.kiv.zcu.cz"  
    ServerName students.kiv.zcu.cz  
    ServerAdmin support@mail.kiv.zcu.cz  
    ErrorLog /var/log/apache2/error_log  
    TransferLog /var/log/apache2/access_log  
    SSLEngine on  
    SSLVerifyDepth 3  
    SSLProtocol All -SSLv2 -SSLv3  
    SSLCipherSuite "EECDH+ECDSA+AESGCM"  
    SSLCertificateChainFile /etc/apache2/ssl/ca-chain.crt  
    SSLCertificateFile /etc/apache2/ssl/keys/kiv.zcu.cz.crt  
    SSLCertificateKeyFile /etc/apache2/ssl/keys/kiv.zcu.cz.pem  
</VirtualHost>
```



2.5 WWW servery – Apache nedostatky

- Robustnost se projevuje na výkonu při od-
bavování velkého množství statického ob-
sahu
- Potřebuje větší množství paměti
- Pokud DNS není při startu nastaveno
správně, zobrazuje obsah prvního virtualu
- Těžko se hledá zdroj přetížení
 - Server-status
 - apachetop
 - Netstat / ss



2.6 WWW servery – Lighttpd, Nginx, Boa,.....

- Menší implementace WWW serverů
- Do určité míry omezené funkce a možnosti
- Menší náročnost na RAM
- Méně propracované API a tedy rychlejší odezva při jednoduchém obsahu
- Typicky distribuce CSS, JS, obrázku atd
- Často se kombinuje s Apache



2.7 WWW servery - IIS

- Internet Information Services
- Dostupné pouze pro MS Windows
- Běžná součást MS Windows 2003, 2008 serverů
- Klikací konfigurace
- Ve světě méně zastoupený než Apache
- Jako „jediný“ podporuje ASP.NET, ale umí i PHP, Perl či Python
- Velmi vhodné pro weby svázané s Active Directory a .NET



2.8 WWW servery - Tomcat

- Robustní WWW server – Java Servlet a JavaServer Pages
- Funguje všude kde je Java
- Specializace na JSP a Servlety
- Velké nároky na HW, hlavně RAM
- Velmi pomalý při startu i zastavení
- Ne úplně přehledná konfigurace
- Webadmin pro správu aplikací



2.9 WWW servery – Jetty, Winstone

- Menší a rychlejší náhrady a Tomcat
- Mírně omezené v možnostech
- Nižší nároky na HW
- Jetty – velmi nezvyklý konfigurační soubor
- Winstone – malý, rychlý přehledný
- Nevhodné pro vše krom JSP a Servletů



3. Programovací jazyky pro WWW

- Původně se WWW skládala jen ze statických HTML stránek
- Časem přibyla možnost SSI – Server Side Includes, dnes už nepoužívané
- Následně první aktivní weby – CGI
 - Většinou skriptovací jazyky Perl či Bash
 - V kompilované podobě pak C či C++
 - Značná bezpečnostní rizika
 - Problém s kompatibilitou vývojového a serverového prostředí



3.1 Programovací jazyky pro WWW

- Dnes nedílná součást každého WWW serveru
- Překládané:
 - ASP.NET
 - Java
 - C
- Interpretované:
 - PHP
 - Perl, Python, Ruby
 - JSP
 - ASP



3.2 Perl

- Dostupné jako modul i jako CGI verze
 - Dříve často používán na CGI
- Velmi mocný nástroj na práci s texty
- Často velmi nepřehledný kód
- I dnes se používá několik velkých projektů v Perlu, například BugZilla



3.3 Perl - příklad

Example: Hello World

- The moment we've all been waiting for!

```
#file:MyApache2/Hello.pm
package MyApache2::Hello;
use strict;
use warnings;
use Apache2::RequestRec ();
use Apache2::RequestIO ();
use Apache2::Const -compile => 'OK';
sub handler {
    my $r = shift;
    $r->content_type('text/plain');
    $r->print('Hello, Apache2/mod_perl World!');
    return Apache2::Const::OK;
}
1;
```



3.4 Perl – příklad 2

Cargo Cult Perl

```
read(STDIN, $buffer, $ENV{CONTENT_LENGTH});
my @pairs = split(/&/, $buffer);
push(@pairs, map { split(/&/, $_) } $ENV{QUERY_STRING});
push(@pairs, map { split(/&/, $_) } @ARGV);
foreach my $pair (@pairs) {
    my ($name, $value) = split(/=/, $pair);
    $name =~ tr/+/ /;
    $name =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",
    hex($1))/eg;
    $value =~ tr/+/ /;
    $value =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/
    eg;
    [... You know the rest... ]
}
```

- Stop the insanity!

Zdroj: <https://www.slideshare.net/oreillymedia/dealing-with-legacy-perl-code-peter-scott>



3.5 Python / Ruby

- Interpretované jazyky
- Dostupné jako modul nebo CGI, ale i jako vlastní server
- Nabývá na oblíbenosti
- Složitější na ladění než PHP
- Projekty
 - Django
 - Redmine



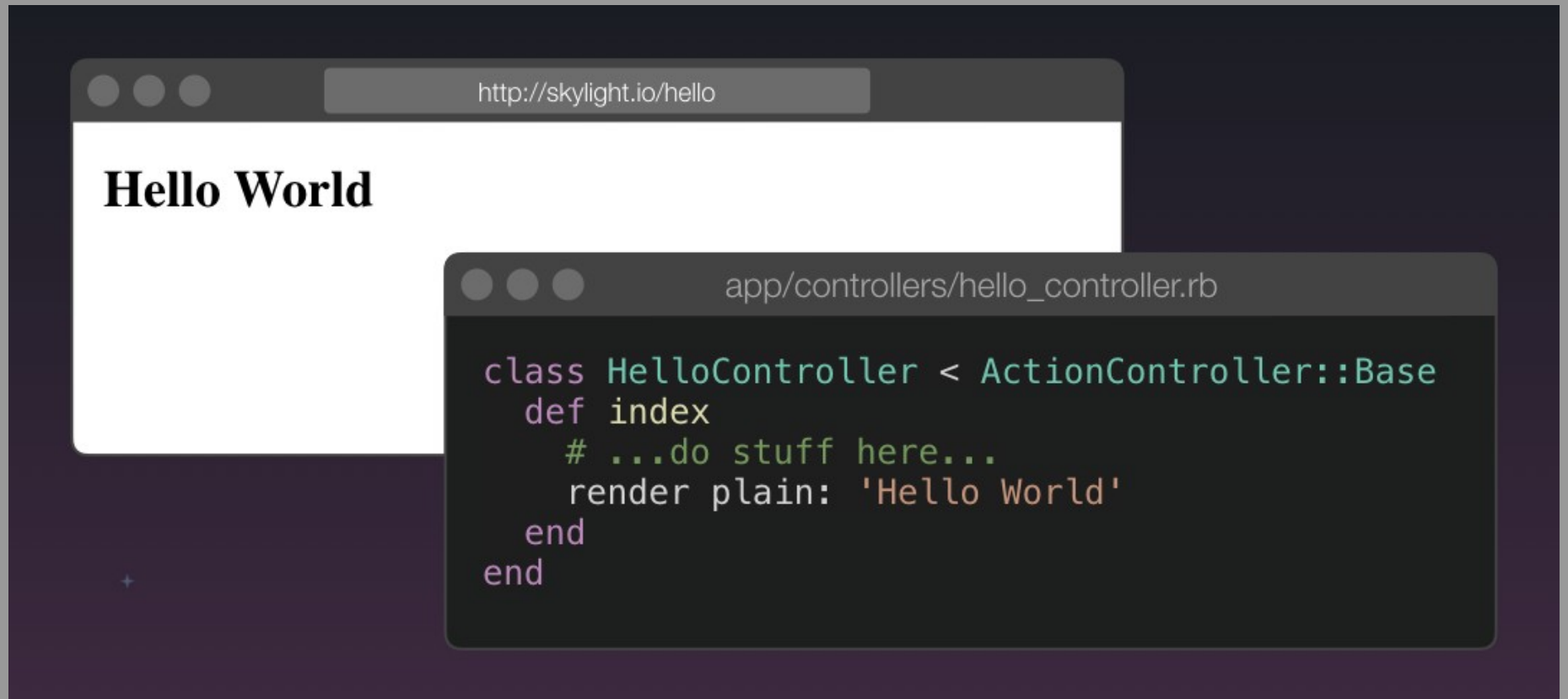
3.6 Python - příklad

```
#!/usr/bin/env python
print "Content-Type: text/html"
print
print """\
<html>
  <head>
    <title>Python - Hello World</title>
  </head>
  <body>
    Hello World
  </body>
</html>
"""
```

Zdroj: <https://www.webhostinghub.com/help/learn/website/how-tos/creating-a-hello-world-python-script>



3.7 Ruby - příklad



Zdroj: <https://blog.skylight.io/content/images/2019/05/railsconf-2019.006-cropped-2.png>



3.8 ASP, ASP.NET

- Obě verze jen pro MS Windows
- ASP je interpretovaná
- ASP.NET překládaná, umí využívat více jazyků z prostředí MS Windows
- Podobný přístup jako pro aplikace v MS Windows
- ASP.NET WebForms – zavádí stavovost do bezstavového protokolu
- Může běžet i samostatně .NET Core
- V Unixu – Mono / .NET Core



3.9 ASP - příklad

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <%Response.Write("Hello World"); %>
    </div>
  </form>
</body>
</html>
```

Response
object to write
Hello World

Zdroj: <https://www.guru99.com/asp-net-first-program.html>

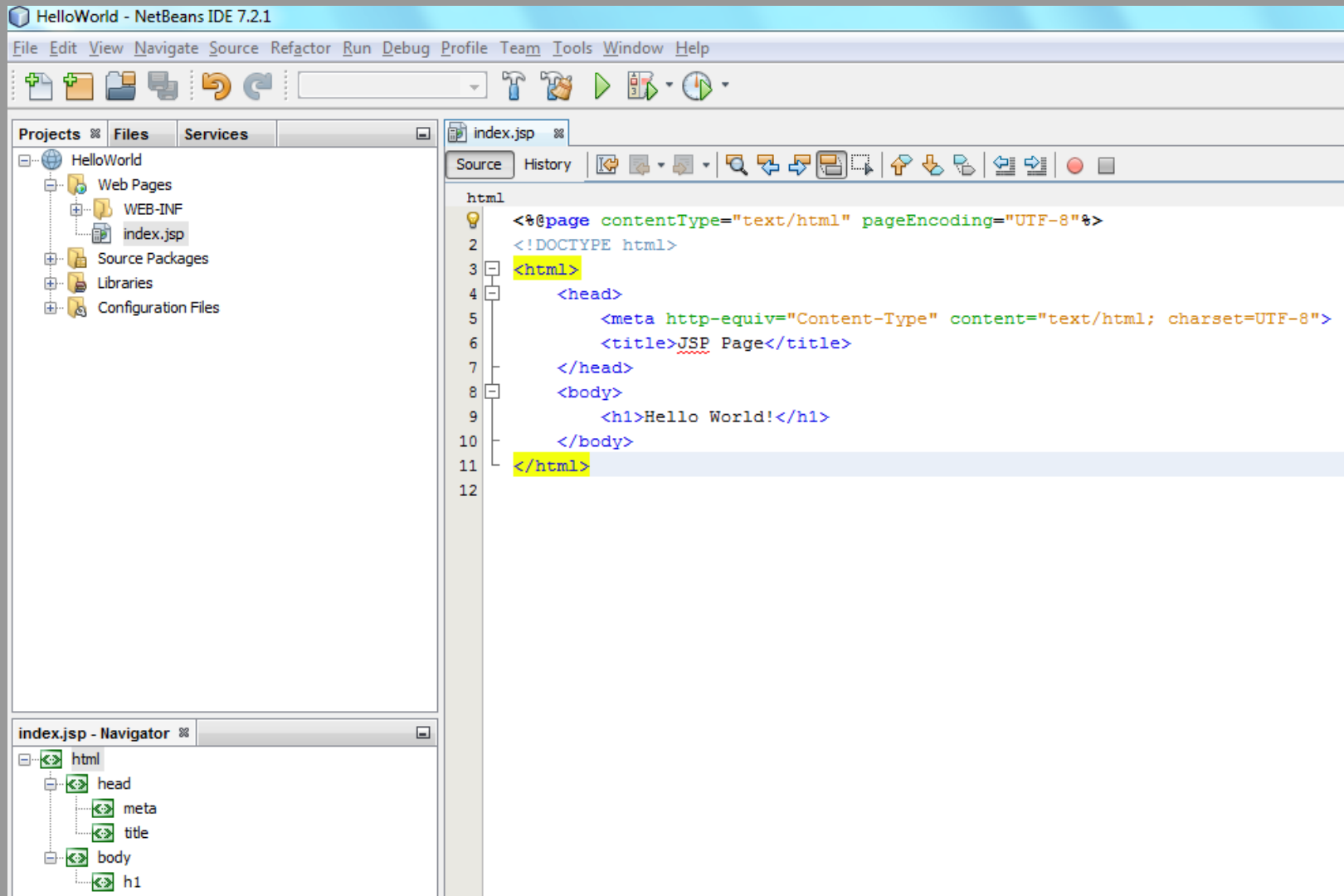


3.10 Java, JSP

- Platformní nezávislost
- Java Servlet – stavový systém
- JSP – interpretovaný
- Potřebuje extra server: Tomcat, Jetty, Winstone
- Velké nároky na RAM i na výkon
- Nesnadné ladění
- Často v kombinaci s proxy jako Apache či Pound



3.11 JSP - příklad



The screenshot shows the NetBeans IDE interface. The main editor window displays the source code of a JSP page named `index.jsp`. The code is as follows:

```
html
1 <%@page contentType="text/html" pageEncoding="UTF-8"%>
2 <!DOCTYPE html>
3 <html>
4   <head>
5     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
6     <title>JSP Page</title>
7   </head>
8   <body>
9     <h1>Hello World!</h1>
10  </body>
11 </html>
12
```

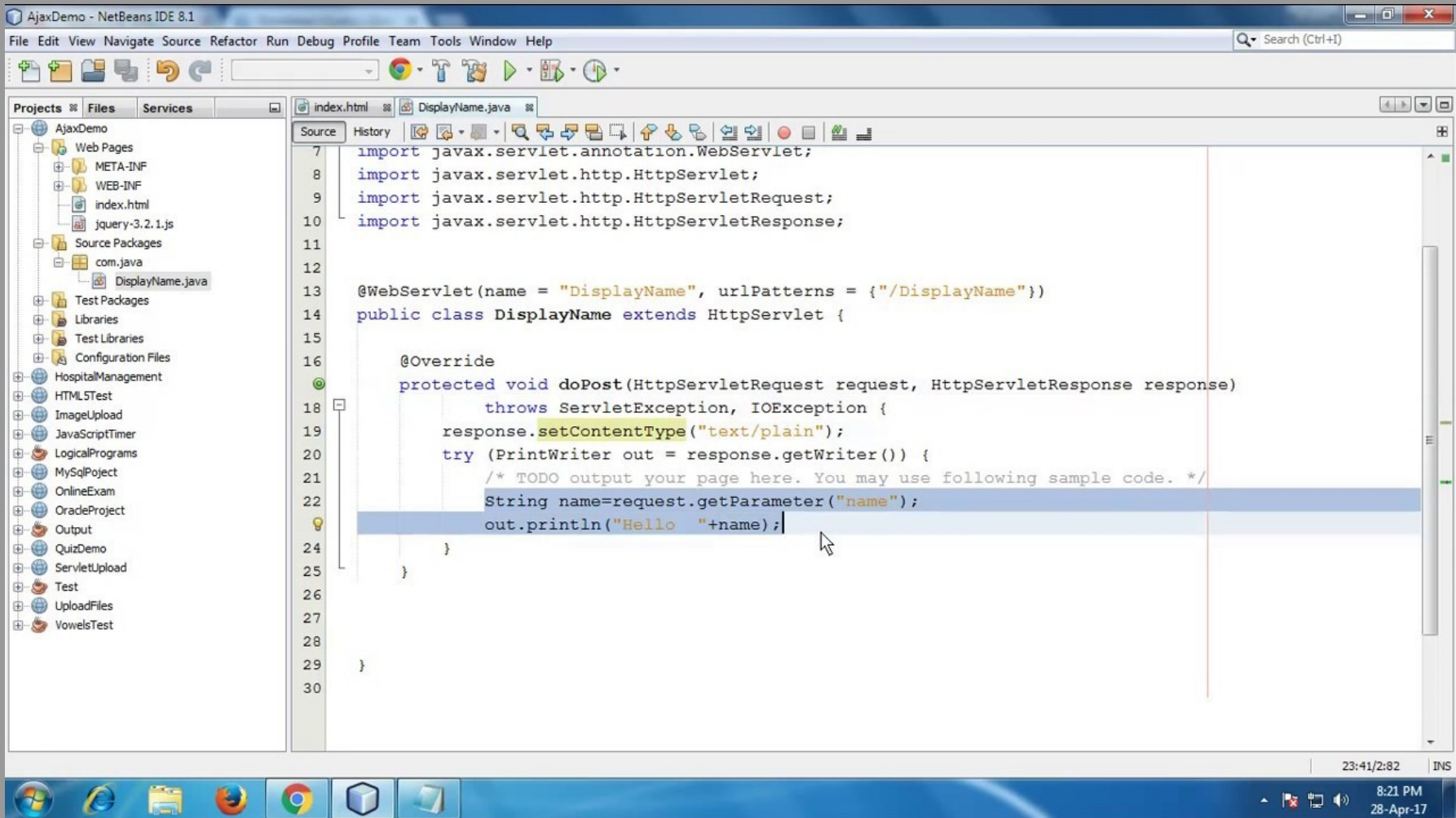
The `index.jsp - Navigator` window at the bottom left shows the following tree structure:

- html
 - head
 - meta
 - title
 - body
 - h1

Zdroj: <https://www.zentut.com/jsp-tutorial/writing-the-first-jsp-page/>



3.12 Java Servlet - příklad



```
7  import javax.servlet.annotation.WebServlet;
8  import javax.servlet.http.HttpServlet;
9  import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11
12
13 @WebServlet(name = "DisplayName", urlPatterns = {"/DisplayName"})
14 public class DisplayName extends HttpServlet {
15
16     @Override
17     protected void doPost(HttpServletRequest request, HttpServletResponse response)
18         throws ServletException, IOException {
19         response.setContentType("text/plain");
20         try (PrintWriter out = response.getWriter()) {
21             /* TODO output your page here. You may use following sample code. */
22             String name=request.getParameter("name");
23             out.println("Hello "+name);
24         }
25     }
26
27
28
29 }
30
```

Zdroj: https://www.youtube.com/watch?v=zY2F_pCSJQ8



3.13 PHP

- Dnes patrně nejrozšířenější programovací jazyk pro WWW
- Aktuální verze 8
 - Ale stále se používají i starší verze 7, 5.6
 - Někdy i 5.2 (poslední zpětně kompatibilní s PHP4)
- Funguje jako FastCGI tak jako modul
- Velmi rychlý vývoj malých a středních projektů
- Dostupné velké množství frameworků a nástrojů



3.14 PHP - příklad

```
1 <html>
2     <head>
3         <title>Page Title</title>
4     </head>
5     <body>
6         <?php
7             echo "Hello World";
8         ?>
9     </body>
10 </html>
```



FIGURE 1: OUTPUT OF EXAMPLE 1

Zdroj: <https://www.btechonline.org/2012/12/php-script-to-print-hello-world.html>




3.15 PHP - konfigurace

- Časté bezpečnostní problémy, většinou chybou programátorů
- Možnost nastavovat parametry pro jednotlivé virtualy i adresáře
- Možnost cachování a optimalizací:
 - Zend, opCache, APC, xCache
- Bezpečnost : **open_basedir, fastcgi**
- Limity:
 - max_execution_time
 - memory_limit
 - upload_max_filesize, file_uploads
 - allow_url_include, allow_url_fopen




3.16 PHPinfo - příklad

PHP Version 7.0.5

System	Windows
Build Date	Mar 30 2016 09:56:03
Compiler	MSVC14 (Visual C++ 2015)
Architecture	x64
Configure Command	<code>cscript /nologo configure.js "--enable-snapshot-build"--enable-debug-pack"--with-pdo-oci=c:\php-sdk\oracle\x64\instantclient_12_1\sdk,shared"--with-oci8-12c=c:\php-sdk\oracle\x64\instantclient_12_1\sdk,shared"--enable-object-out-dir=../obj"--enable-com-dotnet=shared"--with-mcrypt=static"--without-analyzer"--with-pgo"</code>
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\WINDOWS
Loaded Configuration File	C:\Program Files (x86)\EasyPHP-Devserver-16.1\eds-binaries\php\php705vc14x64x160331145413\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20151012
PHP Extension	20151012
Zend Extension	320151012
Zend Extension Build	API320151012,TS,VC14
PHP Extension Build	API20151012,TS,VC14
Debug Build	no
Thread Safety	enabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, zip, compress.zlib, phar
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, mcrypt.*, mdecrypt.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, zlib.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies



Zdroj: <http://rajeshpatadiya.com/php/phpinfo/>



3.15 PHP – OpenSource projekty

- Frameworky
 - Zend, ModX, Symfony, Nette
- Podpůdné nástroje
 - PHPMyAdmin, Adminer
- Administrace webů
 - Joomla, Drupal, Wordpress
 - Administrace webů

