📄 Lab 1: Apache Web Server Installation & Maintenance

Md. Rakibul Kabir [1] **November 8, 2025** [2]

Task 1: Setting up Apache Web Server [3]

Commands for Task 1

Listing 1: Apache Installation Commands [4]

Line	Command	Description
1	# Create Task1 directory	
2	mkdir -p Lab1/Task1 [5]	Creates directory structure.
3	cd Lab1/Task1 [6]	Navigate to the new directory.
4	# Step 1: Install Apache	
5	sudo apt update [7]	Update package list.
6	sudo apt install apache2 [8]	Install Apache web server.
7	# Step 2: Configure Firewall	

8	sudo ufw app list[9]	List available UFW applications.
9	sudo ufw allow 'Apache'[10]	Allow traffic for Apache.
10	sudo ufw status[11]	Check UFW status.
11	# Step 3: Check Web Server	
12	sudo systemctl status apache2[12]	Check Apache service status.
13	# Add webserverlab.com to hosts	
14	echo "127.0.0.1 webserverlab.com" | sudo tee -a /etc/hosts[13]	Map domain to localhost.
15	# Test the web server	
16	curl http://localhost[14]	Test access via localhost.
17	curl http://webserverlab.com[15]	Test access via domain name.

Checkpoint 1[16]Show Apache default page at http://webserverlab.com or http://localhost[17].

Task 2: Setting up Virtual Hosts

Commands for Task 2 - Part 1 (example.com)

Listing 2: Virtual Host Setup for example.com

Line	**Command**	**Description**
1	# Create Task2 directory	
2	mkdir -p Lab1/Task2	Creates directory structure.
3	cd Lab1/Task2	Navigate to the new directory.
4	# Create directory structure for example.com	
5	sudo mkdir -p /var/www/example.com/html	Create document root directory.
6	sudo chown -R \$USER:\$USER /var/www/example.com/html	Change ownership for user access.
7	sudo chmod -R 755 /var/www/example.com	Set directory permissions.

8	# Create index.html for example.com 28	
9	cat > /var/www/example.com/html/index.html << 'EOF' 29	Start creating index.html.
10	<html> 30	HTML start tag.
11	<head> 31	Head section start.
12	<title>Welcome to Example.com</title> 32	Page title.
13	</head> 33	Head section end.
14	<body> 34	Body section start.
15	<h1>Success! The example.com server block is working!</h1> 35	Custom content.
16	</body> 36	Body section end.
17	</html> 37	HTML end tag.

18	EOF 38	End of file input.
19	# Create virtual host configuration 39	
20	sudo tee /etc/apache2/sites-available/example.com.conf << 'EOF' 40	Create VHost config file.
21	<VirtualHost *:80> 41	Start VHost block for port 80.
22	ServerAdmin admin@example.com 42	Administrator email.
23	ServerName example.com 43	Primary domain name.
24	ServerAlias www.example.com 44	Alternate domain name.
25	DocumentRoot /var/www/example.com/html 45	Root directory for the domain.
26	ErrorLog \${APACHE_LOG_DIR}/error.log 46	Error log file.

27	CustomLog \${APACHE_LOG_DIR}/access.log combined 47	Access log file.
28	</VirtualHost> 48	End of VHost block.
29	EOF 49	End of file input.
30	#Enable the site and disable default 50	
31	sudo a2ensite example.com.conf 51	Enable the new VHost.
32	sudo a2dissite 000-default.conf 52	Disable the default site.
33	#Test configuration and restart 53	
34	sudo apache2ctl configtest 54	Check configuration syntax.
35	sudo systemctl restart apache2 55	Restart Apache service.
36	# Add example.com to hosts 56	

37	echo "127.0.0.1 example.com" | sudo tee -a /etc/hosts [57]	Map domain to localhost.

Checkpoint 2 [58]Show http://example.com working with custom content[59].

Testing Different Domains [60]

Listing 3: Testing Domain Routing [61]

- curl http://example.com [62]
- curl http://webserverlab.com [63]
- curl http://127.0.0.1 [64]

Checkpoint 3 [65]Explain what happens when accessing different URLs and why[66].

Commands for Task 2 - Part 2 (anothervhost.com) [67]

Listing 4: Virtual Host Setup for anothervhost.com [68]

Line	Command	Description
1	#Create directory structure for anothervhost.com [69]	
2	sudo mkdir -p /var/www/anothervhost.com/html [70]	Create document root directory.

3	sudo chown -R \$USER:\$USER /var/www/anothervhost.com/html 71	Change ownership.
4	sudo chmod -R 755 /var/www/anothervhost.com 72	Set directory permissions.
5	#Create index.html for anothervhost.com 73	
6	cat > /var/www/anothervhost.com/html/index.html << 'EOF 74	Start creating index.html.
7	<html> 75	HTML start tag.
8	<head> 76	Head section start.
9	<title>Welcome to AnotherVHost.com</title> 77	Page title.
10	</head> 78	Head section end.
11	<body> 79	Body section start.

12	<h1>Success! The anothervhost.com virtual host is working!</h1>	Custom content.
13	<p>This is a different virtual host with unique content.</p>	Additional custom content.
14	</body>	Body section end.
15	</html>	HTML end tag.
16	EOF	End of file input.
17	# Create virtual host configuration	
18	sudo tee /etc/apache2/sites-available/anothervhost.com.conf << EOF'	Create VHost config file.
19	<VirtualHost *:80>	Start VHost block for port 80.
20	ServerAdmin admin@anothervhost.com	Administrator email.

21	ServerName anothervhost.com 89	Primary domain name.
22	ServerAlias www.anothervhost.com 90	Alternate domain name.
23	DocumentRoot /var/www/anothervhost.com/html 91	Root directory for the domain.
24	ErrorLog \${APACHE_LOG_DIR}/anothervhost_error.log 92	Error log file.
25	CustomLog \${APACHE_LOG_DIR}/anothervhost_access.log combined 93	Access log file.
26	</VirtualHost > 94	End of VHost block.
27	EOF 95	End of file input.
28	# Enable the site 96	
29	sudo a2ensite anothervhost.com.conf 97	Enable the new VHost.
30	sudo apache2ctl configtest 98	Check configuration syntax.

31	sudo systemctl restart apache2 [99]	Restart Apache service.
32	#Add to hosts [100]	
33	echo "127.0.0.1 anothervhost.com" | sudo tee -a /etc/hosts 101	Map domain to localhost.

Checkpoint 4 [102]Show http://anothervhost.com working with custom content[103].

Task 3: Hosting Dynamic Websites [104]

Commands for Task 3 [105]

Listing 5: Task 3 Directory Setup [106]

- #Create Task3 directory [107]
- mkdir -p Lab1/Task3 [108]
- cd Lab1/Task3 [109]

Dynamic Website 1: User Registration Form (example.com) [110]

Listing 6: User Registration Form HTML/JavaScript 111

(The following is the content of the HTML/JavaScript file)

HTML

```
<!DOCTYPE html>
<html>
<head>
<title>Example.com User Registration</title>
```

```
<style>
body { font-family: Arial, sans-serif; margin: 40px; }
.container { max-width: 500px; margin: 0 auto; }
input[type="text"], input[type="email"], input[type="password"] {
width: 100%; padding: 10px; margin: 5px 0; border: 1px solid #ddd;
}
input[type="submit"] {
background: #4CAF50; color: white; padding: 10px 20px;
border: none; cursor: pointer; margin-top: 10px;
}
.message { margin-top: 20px; padding: 10px; border-radius: 5px; }
</style>
</head>
<body>
<div class="container">
<h1>Example.com User Registration</h1>
<form onsubmit="return validateForm()">
<label>Full Name: </label>
<input type="text" id="name" required>
<label>Email:</label>
<input type="email" id="email" required>
<label>Password: </label>
<input type="password" id="password" required>
<label>Confirm Password: </label>
<input type="password" id="confirmPassword" required>
<label>Phone Number: </label>
<input type="text" id="phone" required>
<input type="submit" value="Register">
</form>
<div id="message" class="message"></div>
</div>
<script>
function validateForm() {
event.preventDefault();
let name = document.getElementById('name').value;
let email = document.getElementById('email').value;
let password = document.getElementById('password').value;
let confirmPassword = document.getElementById('confirmPassword').value;
let phone = document.getElementById('phone').value;

if (password !== confirmPassword) {
showMessage('Error: Passwords do not match!', 'red');
return false;
}
if (phone.length < 10) {
showMessage('Error: Phone number must be at least 10 digits!', 'red');
return false;
}

showMessage(`Success! Welcome <b>${name}</b>. Registration complete for ${email}. Phone:
${phone}`, 'green');
return false;
}

function showMessage(text, color) {
const messageDiv = document.getElementById('message');
messageDiv.innerHTML = text;
messageDiv.style.color = color;
messageDiv.style.border = '2px solid ${color}';
messageDiv.style.backgroundColor = color === 'green' ? '#e8f5e8' : '#ffe8e8';
}
</script>
</body>
</html>
```

Listing 7: Deploy User Registration Form 112

- #Create dynamic content for example.com 113
- cat > /var/www/example.com/html/index.html << 'EOF' 114
  (Followed by the complete HTML/JavaScript content above)
- EOF 115

Dynamic Website 2: Product Order Form (anothervhost.com) 116

Listing 8: Product Order Form HTML/JavaScript 117

(The following is the content of the HTML/JavaScript file)

HTML

```
<!DOCTYPE html>
<html>
<head>
<title>Another VHost - Product Order</title>
<style>
```

```
body { font-family: Arial, sans-serif; margin: 40px; }
.container { max-width: 600px; margin: 0 auto; }
.form-group { margin: 15px 0; }
label { display: block; margin-bottom: 5px; }
input, select, textarea { width: 100%; padding: 8px; border: 1px solid #ddd; }
button { background: #2196F3; color: white; padding: 10px 20px; border: none; cursor: pointer; }
.order-summary { margin-top: 20px; padding: 15px; border: 1px solid #ddd; }
</style>
</head>
<body>
<div class="container">
<h1>Another VHost - Product Order Form</h1>
<form onsubmit="return processOrder()">
<div class="form-group">
<label>Product Selection: </label>
<select id="product" onchange="updatePrice()">
<option value="">Select a product</option>
<option value="laptop" data-price="1200">Laptop - $1200</option>
<option value="phone" data-price="800">Smartphone - $800</option>
<option value="tablet" data-price="500">Tablet - $500</option>
</select>
</div>
<div class="form-group">
<label>Quantity: </label>
<input type="number" id="quantity" min="1" value="1" onchange="updatePrice()">
</div>
<div class="form-group">
<label>Shipping Address: </label>
<textarea id="address" rows="3" required></textarea>
</div>
<div class="form-group">
<label>Payment Method: </label>
<select id="payment">
<option value="credit">Credit Card</option>
<option value="debit">Debit Card</option>
<option value="paypal">PayPal</option>
</select>
</div>
<button type="submit">Place Order</button>
</form>
<div id="orderSummary" class="order-summary" style="display: none;">
<h3>Order Summary</h3>
<div id="summaryContent"></div>
</div>
```

```
</div>
<script>
function updatePrice() {
const productSelect = document.getElementById('product');
const quantityInput = document.getElementById('quantity');
const selectedOption = productSelect.options[productSelect.selectedIndex];

if (selectedOption.value) {
const price = selectedOption.getAttribute('data-price');
const quantity = parseInt(quantityInput.value);
const total = price * quantity;

document.getElementById('orderSummary').style.display = 'block';
document.getElementById('summaryContent').innerHTML = `
<p><strong>Product:</strong> ${selectedOption.text}</p>
<p><strong>Unit Price:</strong> $${price}</p>
<p><strong>Quantity:</strong> ${quantity}</p>
<p><strong>Total Amount:</strong> $${total}</p>
`;
} else {
document.getElementById('orderSummary').style.display = 'none';
}
}

function processOrder() {
event.preventDefault();
const product = document.getElementById('product').value;
const quantity = document.getElementById('quantity').value;
const address = document.getElementById('address').value;
const payment = document.getElementById('payment').value;

if (!product) {
alert('Please select a product!');
return false;
}

if (!address.trim()) {
alert('Please enter shipping address!');
return false;
}

const selectedOption =
document.getElementById('product').options[document.getElementById('product').selectedIndex
];
const price = selectedOption.getAttribute('data-price');
const total = price * quantity;

alert(`Order placed successfully!\n\nProduct: ${selectedOption.text}\nQuantity:
${quantity}\nTotal: $${total}\nShipping to: ${address}\nPayment: ${payment}`);
document.querySelector('form').reset();
document.getElementById('orderSummary').style.display = 'none';
return false;
}
</script>
</body>
</html
```