Walkthrough

Infrastructure Services for a LAN

Contents

Description

Walkthroughs are intended to bring you through a technical exercise. Most of the steps are provided as a guide to get you started.

A network is just the means to move frames and packets around. To make a network function we need some infrastructural services that are transparent to the end user. In this session, we will look at setting these up in simulation.

Aims

1. To understand the important Infrastructure services common to almost every LAN. Specifically, to simulate;
   a. NTP
   b. DNS
   c. SYSLOG
   d. DHCP
   e. SSH
   f. TFTP

Deliverables

A good idea to keep a list of commands with notes as to why you are using them.

Prerequisites

1. This work was carried out in Cisco Packet Tracer 6.x. You are free to use the simulator of your choice, or real equipment.

Walkthrough

At the core of every network infrastructure are a set of basic services. One of the first things we do when building a green-field site is to build the initial infrastructure server.

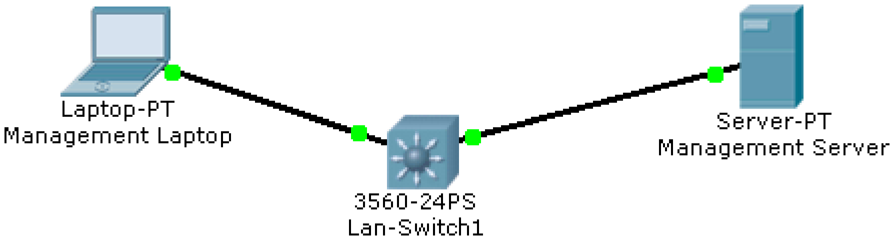
In this session we will get these key services to operate in simulation mode. It's a simulation so the value is limited, but you do get to do a basic configuration and you get to see how the network reacts and is configured to support certain protocols.

On the inside of an Enterprise LAN, most of these services are commonly provided by Windows Server and the interface is relatively easy to use.
For you LAN design DNS and DHCP are sufficient services to include.

Getting Started

To being, open the Packet Tracer file you used for VLANs and call it"NetworkServices".

1. Create a 3560 switch
    a. Call it "Lan-Switch1"
    b. Add the line "ip routing" in the configuration
    c. Create
        i. VLAN21 as Engineering with an interface address of 192.168.21.1
        ii. VLAN22 as Manufacturing with an interface address of 192.168.22.1
    d. Configure Lan-Switch1 with a management IP address of 192.168.10.1 on VLAN10.
    e. Set port fa0/23 and port fa0/24 to be access ports in VLAN10.
2. Create a Laptop and a Server.
    a. Rename the laptop to "Management Laptop", give it an IP address of 192.168.10.254 and a gateway of 192.168.10.1.
    b. Rename the server to "Management Server" , give it an IP address of 192.168.10.253 and a gateway of 192.168.10.1.
3. Link the server to port fa0/24 and the Laptop to port fa0/23, both using a copper straight-through cable.
4. Ensure you can ping the Laptop, Server and Switch.
5. Do a "write memory" command to make sure you save your work! Also save a copy of the file at this point.



6. Implement and test infrastructural network services.

Network Time Protocol (NTP)

Logging is useless without an accurate time stamp. The time stamp is essential for correlation of distributed communication events, forensic analysis, and potential evidentiary use in criminal proceedings.

"….all debugging, security, audit, and authentication is founded on the basis of event correlation (knowing exactly what happened in what order, and on which side), and that depends on good time synchronization. " (from ntp.org).

Many authentication protocols will not work without proper date and time configured.

NTP is a protocol designed to synchronize the clocks of computers over a network to a common time base (usually UTC/GMT). NTP is one of the oldest protocols still in use. To be accurate, it calculates a round trip delay time and offset.

RFC5905-8 establishes the standard NTPv4 for IPv4 and IPv6.

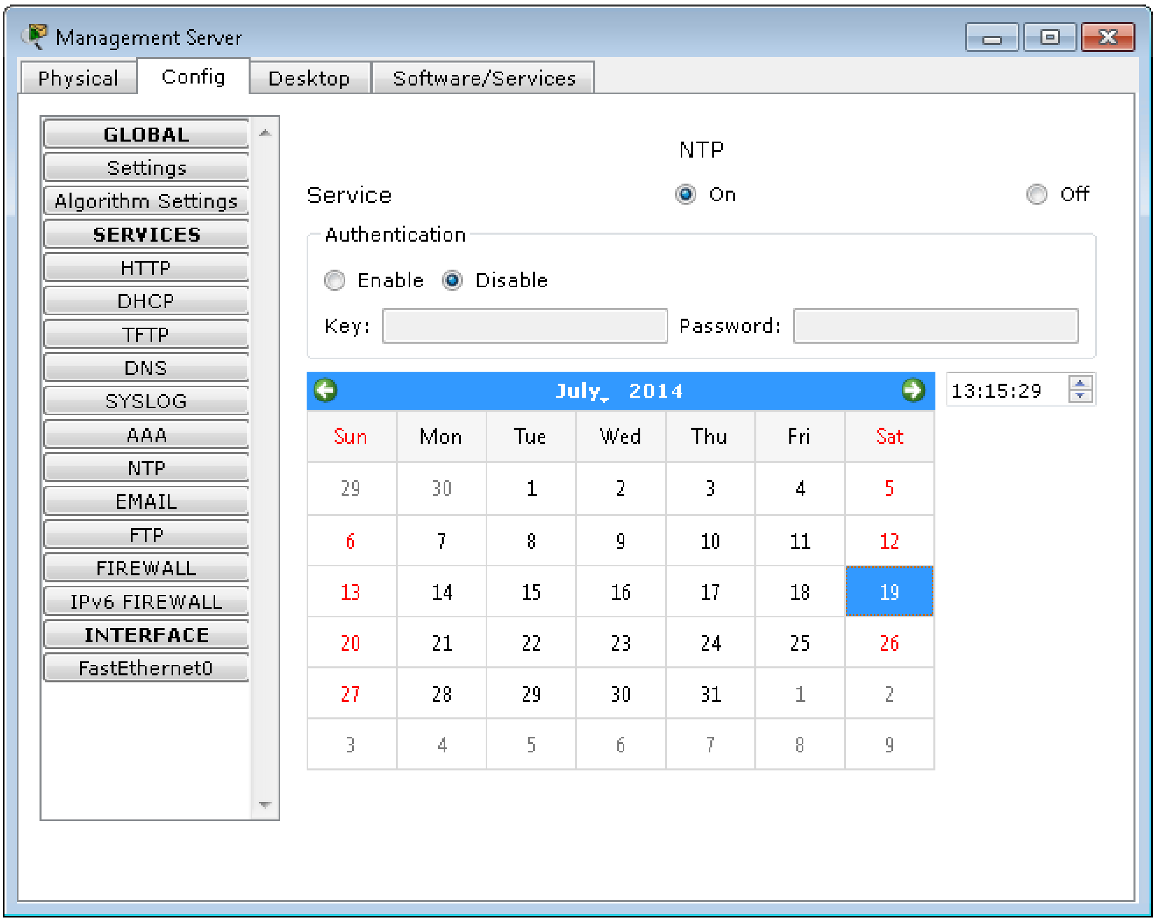
NTP uses UDP Port 123 as standard, you will need to know this when configuring firewalls!

In real world use, we set up NTP as part of a hierarchical network.

On a production network, we will normally set up an NTP server at the earliest stage in commissioning. Every device on the network will synchronize to this server (not quite every device… Microsoft Windows needs to do things a bit differently). We will almost always use a Linux type server for this functionality, for compatibility reaso

Enabling NTP

Go to Management Server and ensure NTP is switched on without authentication.



NTP

Service ● On ○ Off

Authentication

○ Enable ● Disable

Key: Password:

July 2014 13:15:29

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

When you power up a Cisco device it will not have an accurate time and date set. Unusually, there is no real-time clock (RTC) in most Cisco equipment.

To begin, check the current status of the switch. Try the **show clock** command. Also check NTP status as shown.

```
LAN-Switch1#show ntp status
Clock is unsynchronized, stratum 16, no reference clock
nominal freq is 000.0000 Hz, actual freq is 000.0000 Hz, precision is 0**00
reference time is 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1990)
clock offset is 0.00 msec, root delay is 0.00  msec
root dispersion is 0.00 msec, peer dispersion is 0.00 msec.
```

Now tell the switch to get its time from the Management Server.

```
Enter configuration commands, one per line.  End with CNTL/Z.
LAN-Switch1(config)#ntp server 192.168.10.253254
LAN-Switch1(config)#exit
LAN-Switch1#
%SYS-5-CONFIG_I: Configured from console by console

LAN-Switch1#
```

Check to see if this has worked.

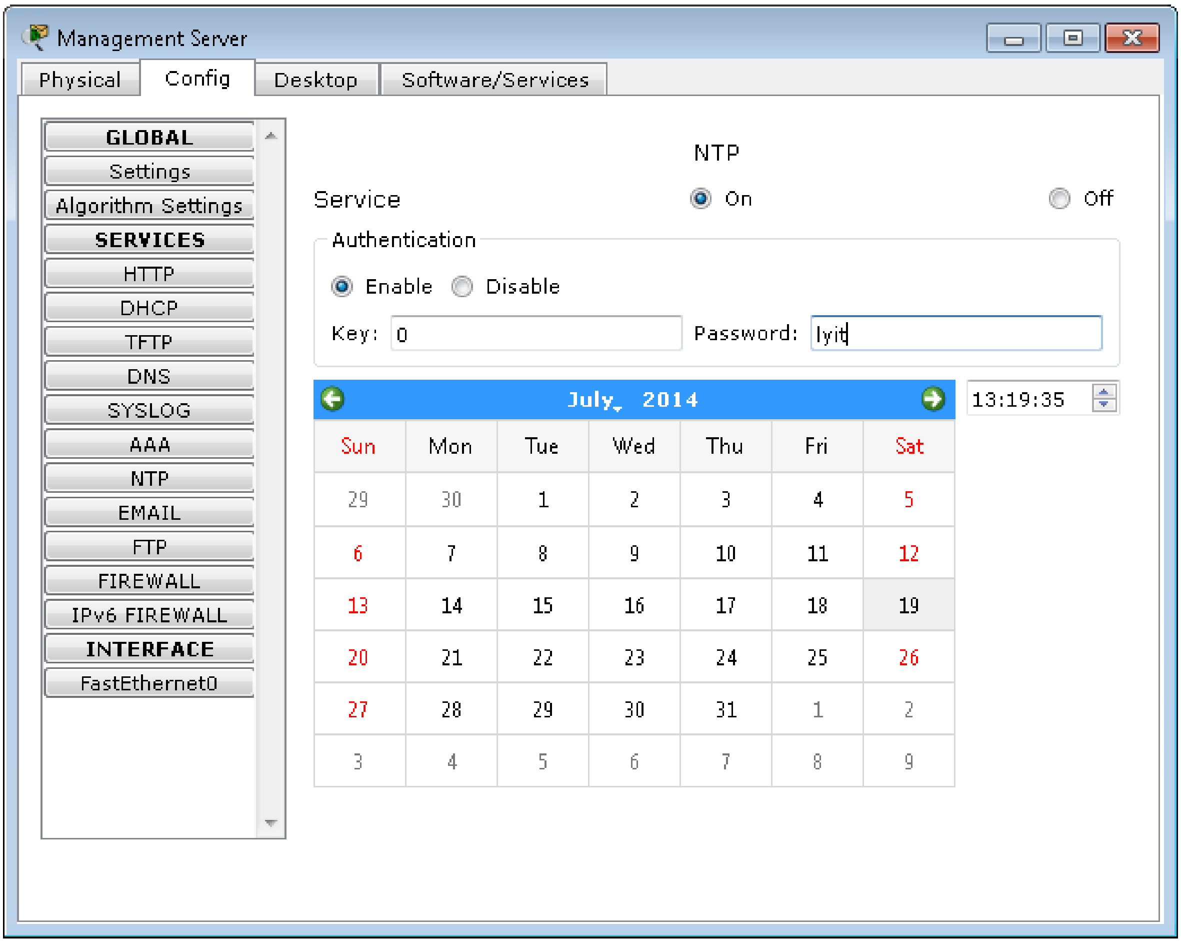
```
LAN-Switch1#show ntp status
Clock is synchronized, stratum 2, reference is 192.168.10.254
nominal freq is 250.0000 Hz, actual freq is 249.9990 Hz, precision is 2**19
reference time is D3BFB2E6.00000144 (13:02:30.324 UTC Tue Aug 28 2012)
clock offset is 0.00 msec, root delay is 0.00  msec
root dispersion is 0.02 msec, peer dispersion is 0.02 msec.
LAN-Switch1#
```

If we wanted to, we could use a device like a switch or router to act as a time server for the rest of the LAN. The C3560 in Packet Tracer does not support this functionality, but many routers do.

In a real world situation you will probably run NTP Server on an internal infrastructure server and/or a Windows Domain Controller.

Securing NTP

Return to the NTP configuration at the Management Server



NTP

Service ● On ○ Off

Authentication

● Enable ○ Disable

Key: 0 Password: lyit

July 2014 13:19:35

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

Define key 0 as being the text “lyit”. At the switch, configure NTP to authenticate.

```
LAN-Switch1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
LAN-Switch1(config)#ntp server 192.168.10.254 key 0
LAN-Switch1(config)#ntp authentication-key 1 md5 lyit
LAN-Switch1(config)#exit
LAN-Switch1#
%SYS-5-CONFIG_I: Configured from console by console

LAN-Switch1#
```

To check to see if everything works, manully set the clock value in the Management Server to two hours into the future. After a few minutes, the switch should update its time. Do a **show run** command to verify and document what you have done. The reset the time to the actual!

Domain Name Services (DNS)

Computers like to use binary, people don't. For the Internet to work, we use its equivalent of a phone book, something which can look up numbers to give names and vice versa.

DNS is a hierarchical distributed database and is one of the components of the internet about which severer security concerns now exist, without a great deal of solutions on the horizon. DNS was standardized in RFC1035 but a whole host of additional RFCs have been developed and the standard is in constant evolution.

In DNS, a phone book entry is called a resource record (RR). A resource record has the following fields.

NAME: Owner name, the name of the node to which this resource record pertains.
TYPE: Two octets containing one of the RR type codes.
CLASS: Two octets containing one of the RR class codes.
TTL: A 32 bit signed integer that specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
RDLENGTH: An unsigned 16 bit integer that specifies the length in octets of the RDATA field.
RDATA: the payload!

NAME	Owner name, the name of the node to which this resource record pertains.
TYPE	Two octets containing one of the RR type codes.
CLASS	Two octets containing one of the RR class codes.
TTL	A 32-bit signed integer that specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
RDLENGTH	An unsigned 16-bit integer that specifies the length in octets of the RDATA field.
RDATA	The payload!

The following are typical resource records codes for common services.

A	An IPv4 Address
AAAA	An IPv6 Address
NS	A Name server which is authorative for the domain
CNAME	An alias
PTR	A pointer, used for reverse lookup
MX	Mail exchanger

In this practical session we will set up a simple DNS for internal infrastructure usage. This will normally be the second service we will bring live on a real network, after NTP and before we get any client services operational.

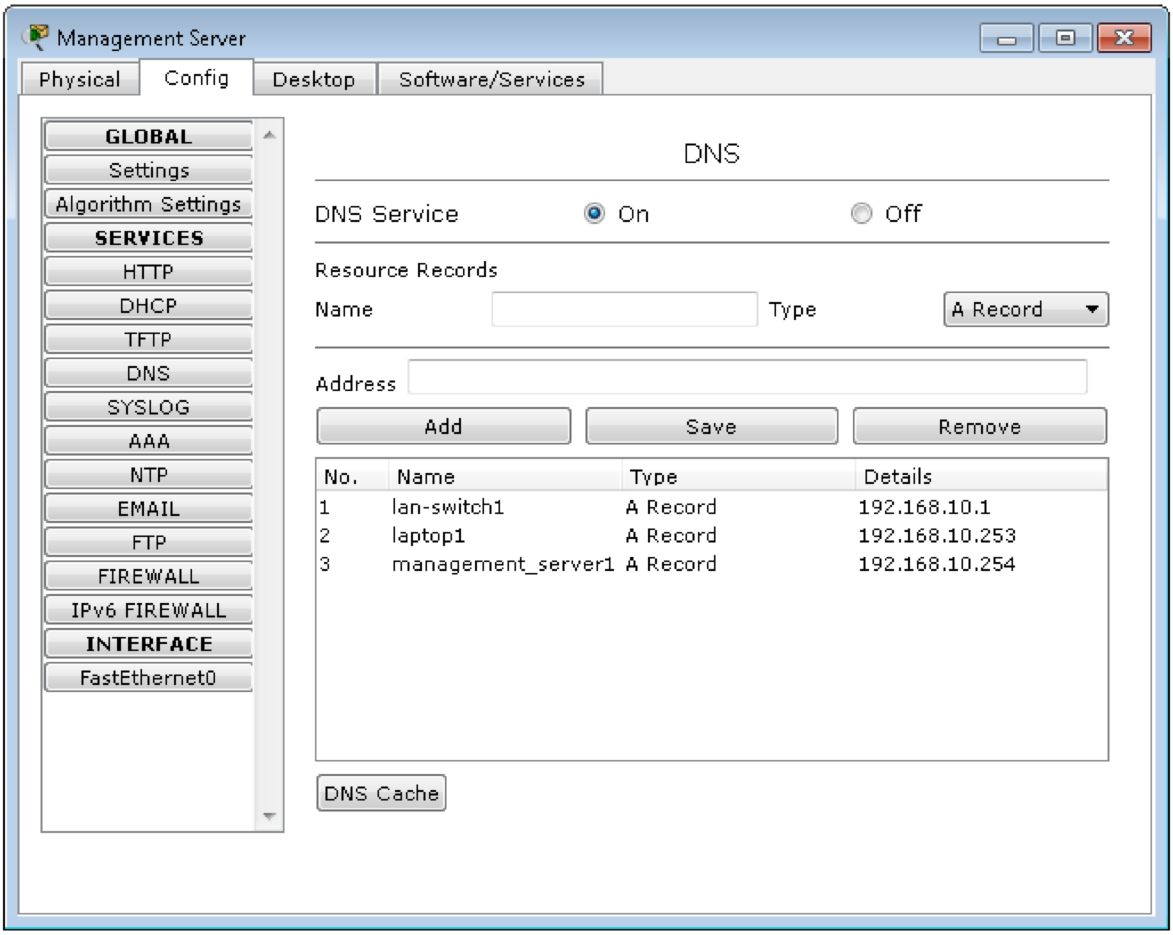
For detailed information on DNS, read "DNS and Bind" published by O'Reilly.

In a real network, we almost always use Linux and BIND for public DNS and Active Directory integrated Windows for internal DNS.

. BIND is the reference implementation for DNS and is the most accurate and standard software for DNS. Take a look at https://www.isc.org/downloads/bind/

Enabling DNS

Go to the Management Server and ensure DNS server is switched on.



DNS

DNS Service ● On ○ Off

Resource Records

Name Type A Record

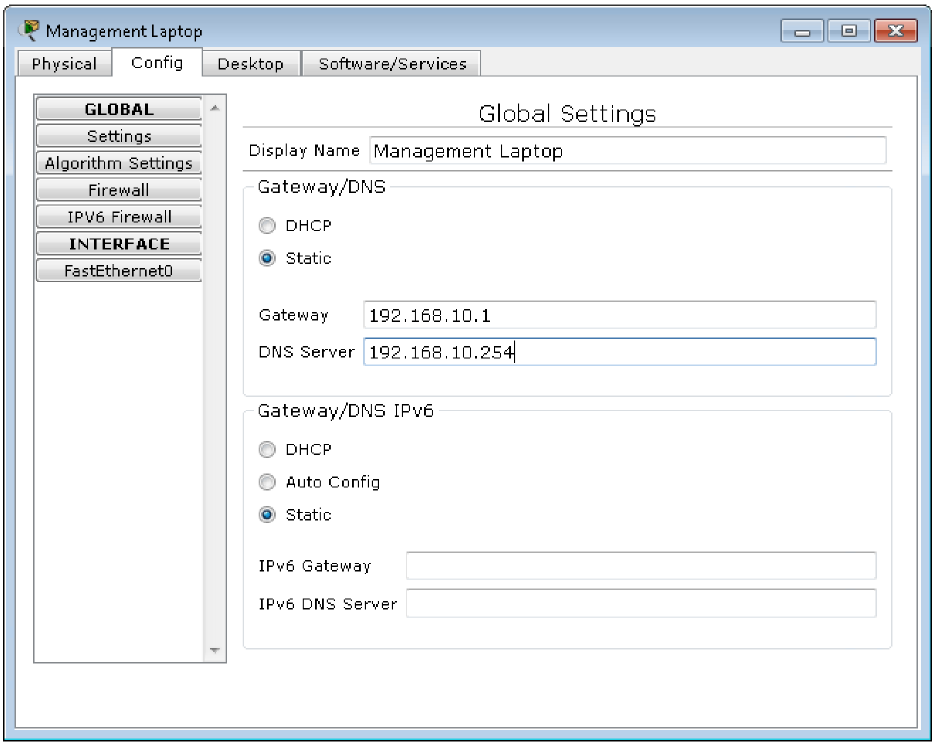
Address

No.	Name	Type	Details
1	lan-switch1	A Record	192.168.10.1
2	laptop1	A Record	192.168.10.253
3	management_server1	A Record	192.168.10.254

Add records for the three devices we have (use A records). Real DNS is a bit more complex, we would add in reverse records, have a full domain name, add in NS records etc.

Testing DNS from Clients

Go to the Management Laptop and configure the DNS server. The gateway off the network is the address of Lan-Switch1 (because it's a L2/L3 switch).



Global Settings

Display Name: Management Laptop

Gateway/DNS

○ DHCP
● Static

Gateway: 192.168.10.1

DNS Server: 192.168.10.254

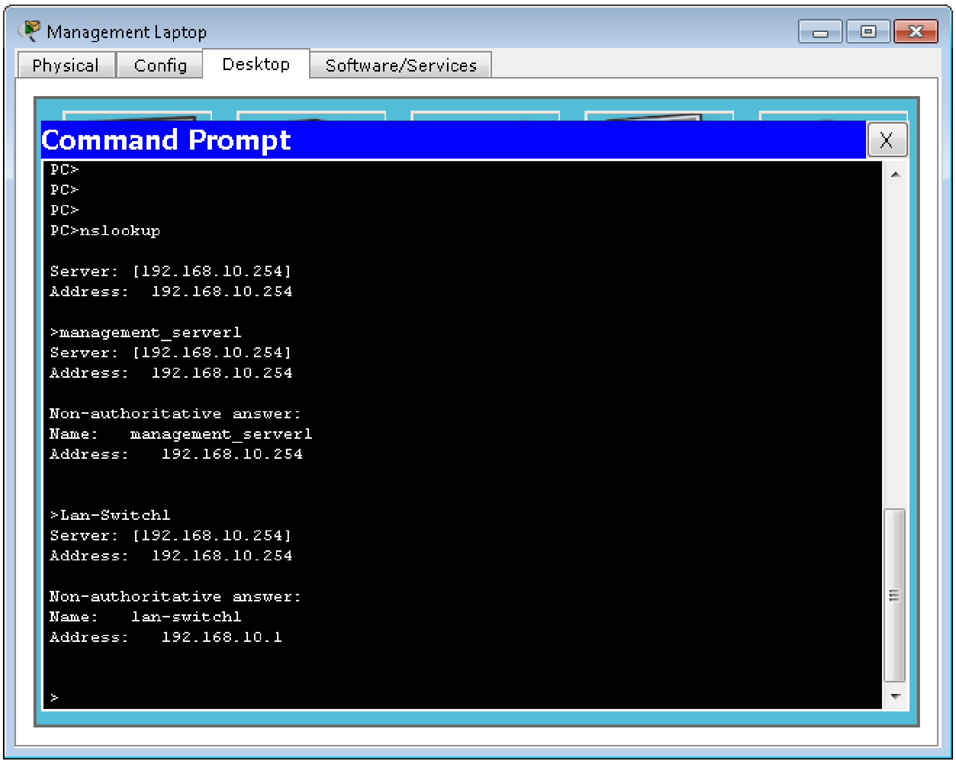
Gateway/DNS IPv6

○ DHCP
○ Auto Config
● Static

IPv6 Gateway:

IPv6 DNS Server:

Now we can attempt an **nslookup** from the command prompt to see if DNS works.



```
PC>
PC>
PC>
PC>nslookup

Server: [192.168.10.254]
Address:  192.168.10.254

>management_server1
Server: [192.168.10.254]
Address:  192.168.10.254

Non-authoritative answer:
Name:   management_server1
Address:   192.168.10.254


>Lan-Switch1
Server: [192.168.10.254]
Address:  192.168.10.254

Non-authoritative answer:
Name:   lan-switch1
Address:   192.168.10.1


>
```

Testing DNS from the switch

Go to the switch and do a **show run** command.

Enter configuration mode and type the commands

```
ip name-server 192.168.10.254
ip domain lookup
```

Then exit configuration mode. Use the command **show run** to check your configuration. Now do a ping test to see if the names resolve. The first ping always fails, don’t worry about it. This is all to do with the way ARP works.

```
LAN-Switch1#ping laptop1
Translating "laptop1"...domain server (192.168.10.254)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.253, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 20/25/31 ms

LAN-Switch1#
```

SYSLOG Protocol

SYSLOG had its origins in UNIX and was intended for logging system messages. It has been standardized in RFC3164. The protocol provides a transport to allow a device to send event notification messages across IP networks to SYSLOG servers. The protocol is designed to transport these event messages from the generating device to the collector. The collector doesn't send back an acknowledgment of the receipt of the messages.

It has become a heterogeneous network logging tool which works with almost everything. Even cheap SOHO devices will generally have SYSLOG and all devices we might use on an enterprise network will certainly have it. Syslog is both a communications protocol and a set of programs and libraries.

On a real network, we will initially set up SYSLOG on a Linux server; cheap and easy. However, we need to be careful. It is possible to have usernames (and passwords!!) and other critical information flying around in the open from SYSLOG clients (e.g. almost every device) to the central SYSLOG server.

Once the network gets bigger or for enterprise networks, we will probably use a dedicated package.

SYSLOG Messages

SYSLOG messages have a standard format: Date, Time, System, Facility: Message

Any Linux server can act as a network syslog server, aggregating syslog messages. On an initial network build, we will almost always use a Linux server as our first logging device. As the network builds in size and complexity, we may use specialised SYSLOG software and reporting tools or perhaps a dedicated appliance. SYSLOG uses port UDP or TCP port 514, almost always UDP. The packet structure is detailed in RFC5424.

A SYSLOG packet will normally have the following fields of data.

Facility

The sender component/application that sent the message, typically;
On CISCO equipment, we have facilities called local0, local1, local2, local3, local4, local5, local6, and local7. By default, messages are sent as local7.

Severity

0 = Emergencies
1 = Alerts
2 = Critical
3 = Errors
4 = Warnings
5 = Notifications
6 = Informational
7 = Debugging

Timestamp (guess why we configured NTP first!!).

Host

Ideally will be a hostname not an IP address.

Tag

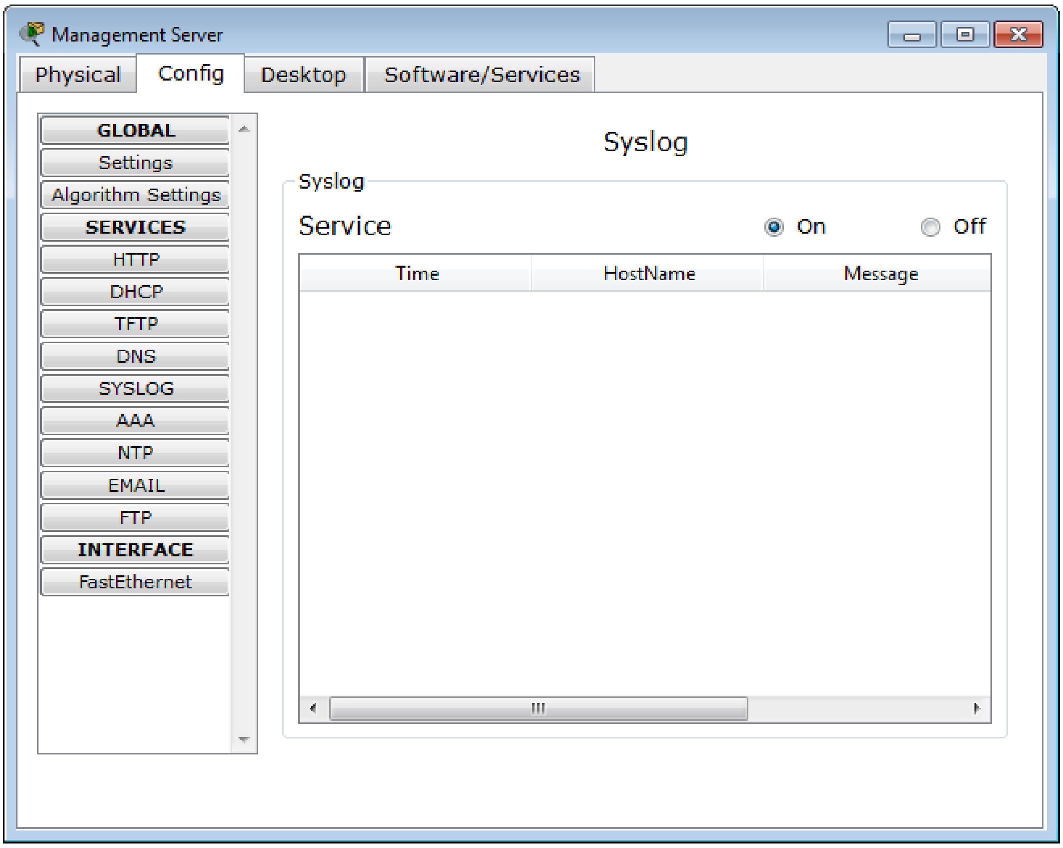
The use of which varies greatly between applications, but it may contain the name of the originating process or device.

Message

The message field may contain machine readable fields or unstructured text.

Enabling SYSLOG

Go to the Management Server and ensure SYSLOG is switched on.



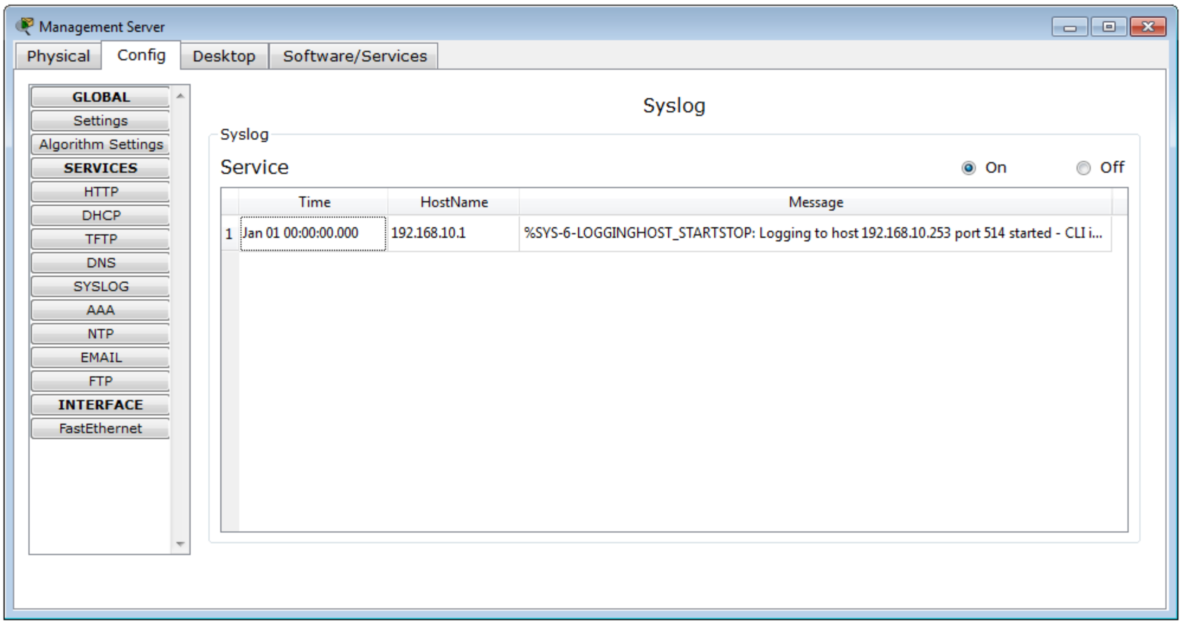
Service ● On ○ Off

Time	HostName	Message

We need to turn on logging at the network switch.

```
LAN-Switch1>ena
LAN-Switch1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
LAN-Switch1(config)#logging ?
  A.B.C.D   IP address of the logging host
  buffered  Set buffered logging parameters
  console   Set console logging parameters
  host      Set syslog server IP address and parameters
  on        Enable logging to all enabled destinations
  trap      Set syslog server logging level
  userinfo  Enable logging of user info on privileged mode enabling
LAN-Switch1(config)#logging on
LAN-Switch1(config)#logging 192.168.10.254
LAN-Switch1(config)#%SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.10.254 port
514 started - CLI initiated
```

If you now return to the server, Config->SYSLOG you should see that logging is taking place.



Service ● On ○ Off

	Time	HostName	Message
1	Jan 01 00:00:00.000	192.168.10.1	%SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.10.253 port 514 started - CLI i...

SYSLOG Detailed Configuration

Now that we are working, we can get a bit more sophisticated. Firstly we can set the trap level, which allows us to select messages at a particular severity level.

If I type in the command **logging trap level 4** then only syslog messages from warnings upwards will be forwarded. We can use this granularity to reduce nuisance alerting and allow us to see the important stuff. We need to be careful. Set the level too high and you miss early warnings that things are going wrong. Set it too low and you end up flooded with logs which you never check.

We can also set the logging facility using the command **logging facility localx** where x is a number from 0-7.

CISCO devices are quite limited in memory. We can set the device up to buffer SYSLOG messages locally using the command **logging buffered y** where y is the buffer size. Typically use 4096.

Not all of the above will work in Packet tracer.

Securing Logs

SYSLOG does not contain any important data so you don't really care about the log files? WRONG. In the event of a breach, they are your first port of call. Also in the event of an attack on your network, the first thing any hacker will want to do is to erase the log files.

We will generally keep the SYSLOG server on a firewalled subnet with only UDP port 514 exposed.

Dynamic Host Configuration Protocol (DHCP)

Manually configuring computers attached to a network is a time-consuming and error-prone process. Dynamic Host Configuration Protocol is an automatic configuration protocol used on IP networks. DHCP allows a computer to be configured automatically eliminating the need for intervention by a network administrator. It provides a central database for keeping track of computer's IP allocation, this prevents two computers from accidentally being configured with the same IP address.

DHCP provides one of our key diagnostic and management information sources for a LAN and may integrate with other services (DNS, WINS) to provide an integrated and "joined up thinking" approach.

Computers must be configured with specific IP information before they can communicate with other computers even on their own subnet. At a minimum they need;

- IP Address
- Subnet Mask

But the point of layer 3 protocols like IP is to communicate with computers on other subnets. Computers must be configured with a gateway address before they can communicate with other computers on another subnet
Computer may also be configured with a range of other essential information such as;

- DNS Servers
- NTP Servers
- WINS

Planning DHCP Scopes

This practical will deploy a very simple DHCP server in Packet Tracer. We normally use fixed IP addresses for infrastructure, so servers, switches and routers tend to not use DHCP. We may provide DHCP addresses to infrastructure subnets for ease of configuration. If we are going to do this, it is always useful to define our internal rules. We would normally do this in network planning before we power on the first device of our new network.

A typical rule set might look like this.

For each subnet, keep a reserved range

- For infrastructure
- For clients via DHCP
- For clients and servers on fixed addresses

Subnet Mask

Subnet	Mask	Gateways	Infrastructure	Dynamic Clients	Fixed Clients
/24	255.255.255.0	1, 2, 3, 4, 5, **20**	21-99	100-199	200-254

In a typical class C network;

We will keep the last octet 1-5 for physical routers. If we are using redundancy or some form of failover, we may have more than one real physical gateway. We will get these routers to masquerade as .20 so the gateway address in every case will be .20.

We will keep addresses 21-99 for fixed infrastructure in that subnet. This could be used for switches, radio APs, and other infrastructure devices.

We will assign 100 addresses for clients, to be allocated dynamically by DHCP.

We will keep the addresses from 200-254 for servers and clients which require fixed addresses. We can document these in DHCP if we want, we can even assign them from DHCP, but they are fixed addresses.

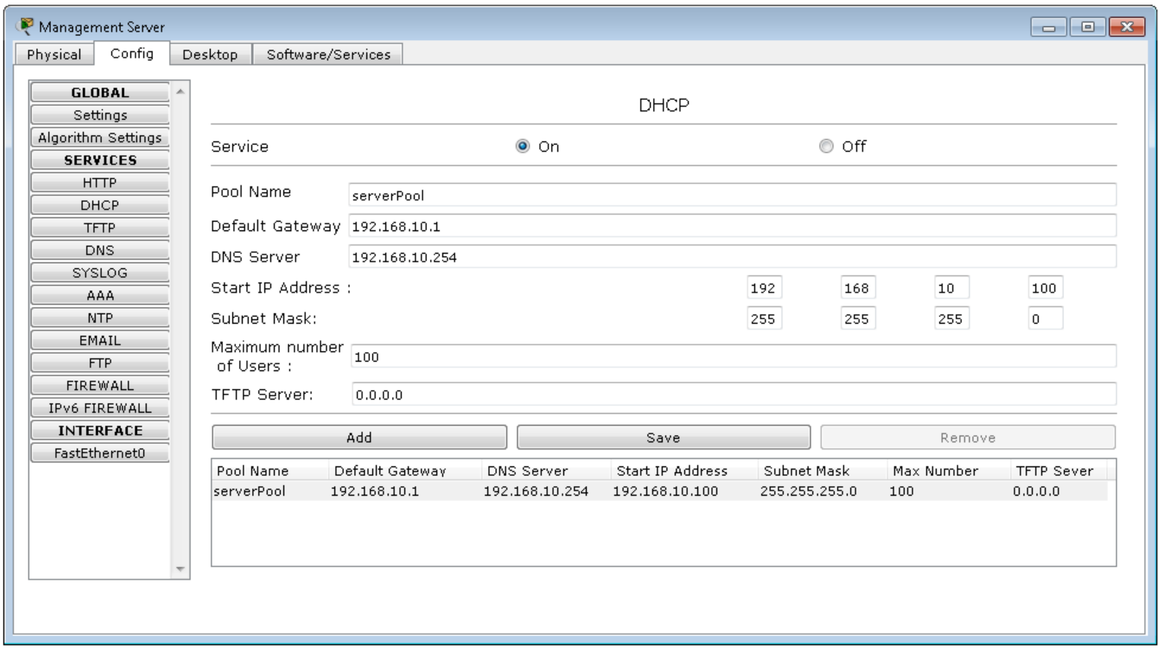
Subnet	Mask	Gateways	Infrastructure	Dynamic Clients	Fixed Clients
>/24	Varies	In the First Range 1, 2, 3, 4, 5, **20**	First Range of 255 addresses 21-99	2nd to Penultimate Range	Last Range

For LANs we commonly use RFC1918 address space. If you do not know what this is, you need to do an Internet search and fully understand the implications of RFC1918. One of the benefits of RFC1918 is that there is no shortage of addresses. For this reason, we can build in quite a bit of overhead, as we have done in these examples.

If you have to use public IP addresses, you will probably be much more precise in allocating only what is required to each function. This requires quite a bit of detailed planning, as it can be very difficult to re-engineer an IP addressing plan after it has been implemented.

Configuring DHCP

Go to the Management Server and select DHCP. We will now create a pool of addresses that conforms to our rules for a class C network, as discussed earlier.



DHCP

Service ● On ○ Off

Pool Name serverPool

Default Gateway 192.168.10.1

DNS Server 192.168.10.254

Start IP Address : 192 168 10 100

Subnet Mask: 255 255 255 0

Maximum number of Users : 100

TFTP Server: 0.0.0.0

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max Number	TFTP Sever
serverPool	192.168.10.1	192.168.10.254	192.168.10.100	255.255.255.0	100	0.0.0.0

Press **Add** to include this pool in the DHCP server.

Now to test DHCP! Go to the network management laptop. Go to **Config->Fast Ethernet** and change the IP configuration to DHCP. Now go to **Laptop1->Desktop->Command Prompt** and type **ipconfig** to see if your laptop has updated correctly. When you are done, change the laptop back to its original static IP address.

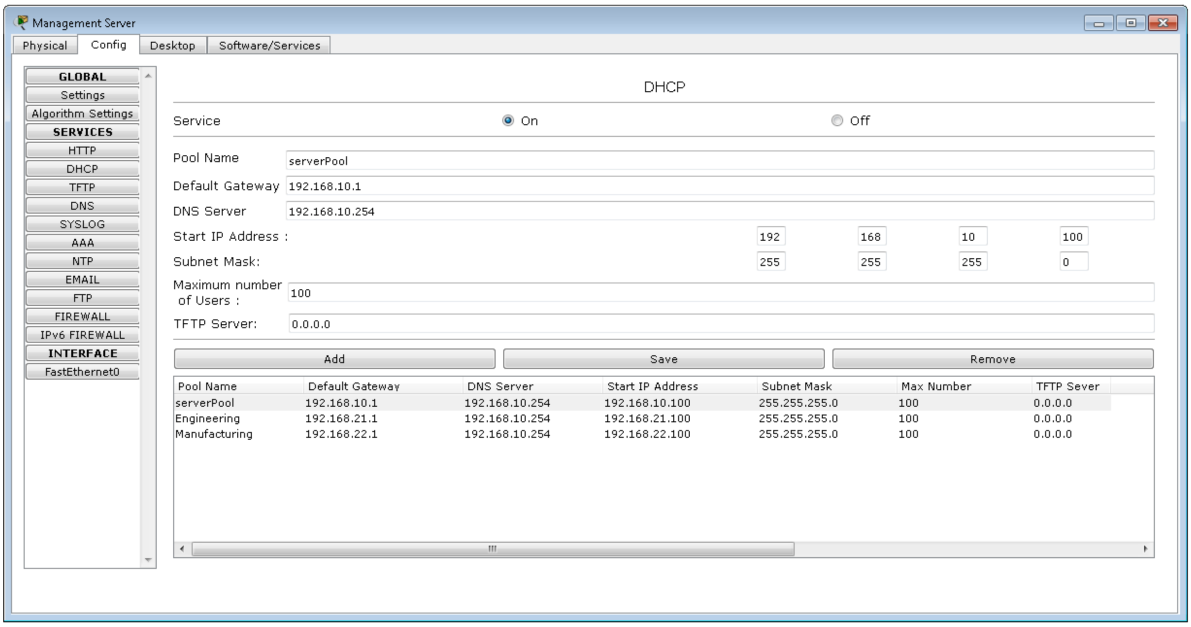
So now we have DHCP working on a subnet which is only intended for infrastructure and management devices.

Our clients are on completely different subnets and even if they had some magic way of knowing about the server, they can't get off the network they are on (because they don't have a layer 3 address.

We need some magical device which can somehow route Ethernet packets to a DHCP server, despite that fact that the source of these packets (a client PC) does not have an IP address. We have such a magic function, it's called DHCP forwarding on a router and it requires the router to explicitly know for each VLAN or subnet, where to forward DHCP packets.

Now create two client PCs.

1. Create a PC called Engineering1____________ 2. Connect it to fa0/1an appropriate port. 3. Make fa0/1 an access port in VLAN21 4. On the Management Server, create an appropriate DHCP pool for this VLAN	1. Create a PC called Manufacturing1____________ 2. Connect it to fa0/2an appropriate port. 3. Make fa0/2 an access port in VLAN22. 4. On the Management Server, create an appropriate DHCP pool for this VLAN. My configuration is shown below.



Service ● On ○ Off

Pool Name: serverPool

Default Gateway: 192.168.10.1

DNS Server: 192.168.10.254

Start IP Address : 192 168 10 100

Subnet Mask: 255 255 255 0

Maximum number of Users : 100

TFTP Server: 0.0.0.0

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max Number	TFTP Sever
serverPool	192.168.10.1	192.168.10.254	192.168.10.100	255.255.255.0	100	0.0.0.0
Engineering	192.168.21.1	192.168.10.254	192.168.21.100	255.255.255.0	100	0.0.0.0
Manufacturing	192.168.22.1	192.168.10.254	192.168.22.100	255.255.255.0	100	0.0.0.0

We need to add commands to the interface to tell any client on the Engineering or Manufacturing VLAN where to find the DHCP server (which in our case is 192.168.10.254). Try the following commands.

```
Lan-Switch1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Lan-Switch1(config)#int vlan 21
Lan-Switch1(config-if)#ip helper-address 192.168.10.254
Lan-Switch1(config-if)#exit
Lan-Switch1(config)#int vlan 22
Lan-Switch1(config-if)#ip helper-address 192.168.10.254
Lan-Switch1(config-if)#exit
Lan-Switch1(config)#
```

Exit to get back to the # prompt and do a **show run** to make sure everything looks right!

Now go to the command prompt on each of Engineering1 and Manufacturing1, use ipconfig to verify you have good DHCP information and then try to ping around the network. You should be able to get to everything. If you still don't have an IP address, go to the command prompt and type **ipconfig /release** followed by **ipconfig /renew** to get the test PC to sort itself out.

DHCP Security Considerations

Every modern LAN that is bigger than a handful of clients uses DHCP. A typical host sends out a DHCP request and the first server to respond will supply IP Address, mask, gateway, DNS and anything else required.

DHCP snooping is a security feature that filters untrusted DHCP messages and maintains a DHCP snooping binding database. Do some reading on DHCP Snooping.

Secure Shell (SSH)

Early network applications all used telnet to remotely access devices. We should never use telnet in any modern real-world network. It is unsecure and anyone capable of examining network traffic can extract everything which traverses the network, including passwords! In general, you should configure all UNIX servers and network devices to be accessed using SSH and should disable telnet.

SSH allows secure access to a shell interface across public networks. It uses public key cryptography to authenticate the remote computer. SSH was developed in 1995 by Tatu Ylonen of the Helsinki University of Technology in response to a password sniffing attack.

SSH Communications Security was founded to commercialize and support SSH, this company is now called Tectia. The Tectia client and server software is still a good enterprise choice and is commonly used.

Many free versions are in common use, Putty being the one most used at LYIT.

Configuring SSH

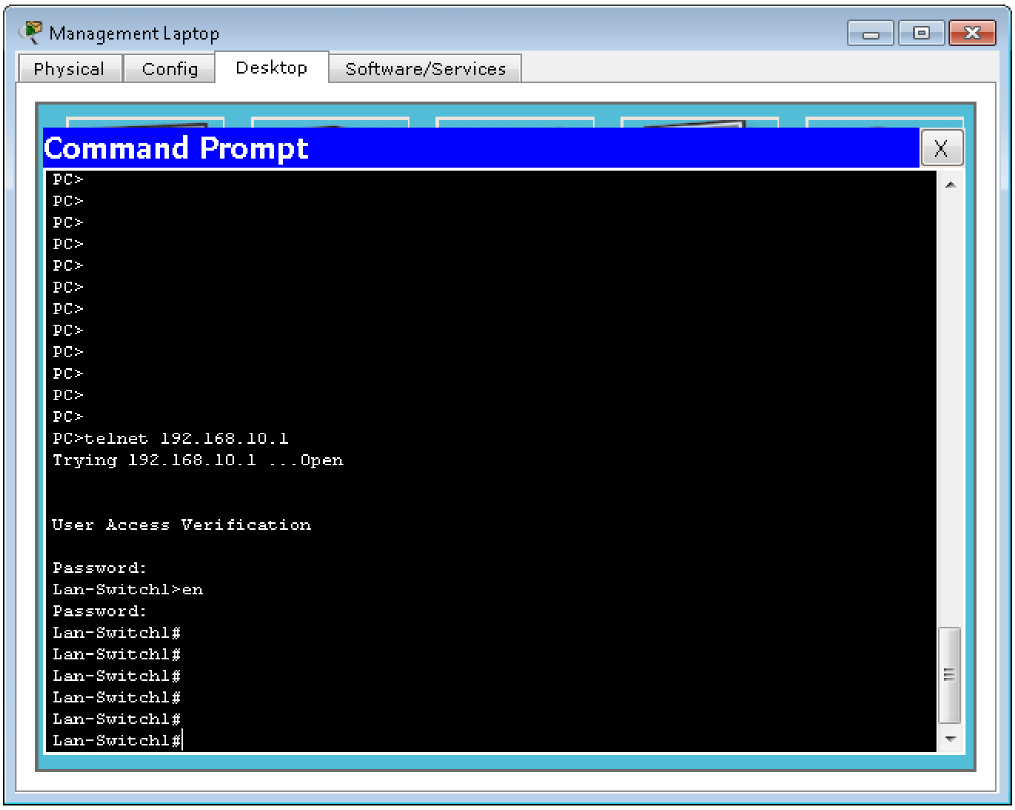
Way back at the start of this practical sequence, we set passwords on the console. You may have decided to skip this step or may need to revisit it. Unless you have specifically set a telnet password, you will not be able to login to a switch remotely. On Lan-Swich1, configure telnet access, use the following commands.

```
line vty 0 4
exec-timeout 5 30
 password MyTelnet
 logging synchronous
 login
 history size 10
exit
```

Now test to make sure you can login. Go to Laptop1 ping and then telnet to 192.168.10.1. Now try an **enable** command. If this does not work, you need to carry out the following command, also from the first practical!

```
enable secret MyPassword
```

Now we have successfully created a completely insecure and dangerous way to login to our network devices.



```
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>telnet 192.168.10.1
Trying 192.168.10.1 ...Open


User Access Verification

Password:
Lan-Switch1>en
Password:
Lan-Switch1#
Lan-Switch1#
Lan-Switch1#
Lan-Switch1#
Lan-Switch1#
Lan-Switch1#
```

Try to login to the switches by using SSH from Laptop1; it will fail because we have not yet set it up. We need to fix that now and configure SSH. To do so, we have some pre-requisites we need to configure. First, we need to set the domain name of the domain the router is in.

```
ip domain-name test.lyit.ie
```

We need to check to see if cryptographic keys have already been generated on this device. Type the command

```
show crypto key mypubkey rsa
```

This will probably return blank. If so, generate public keys.

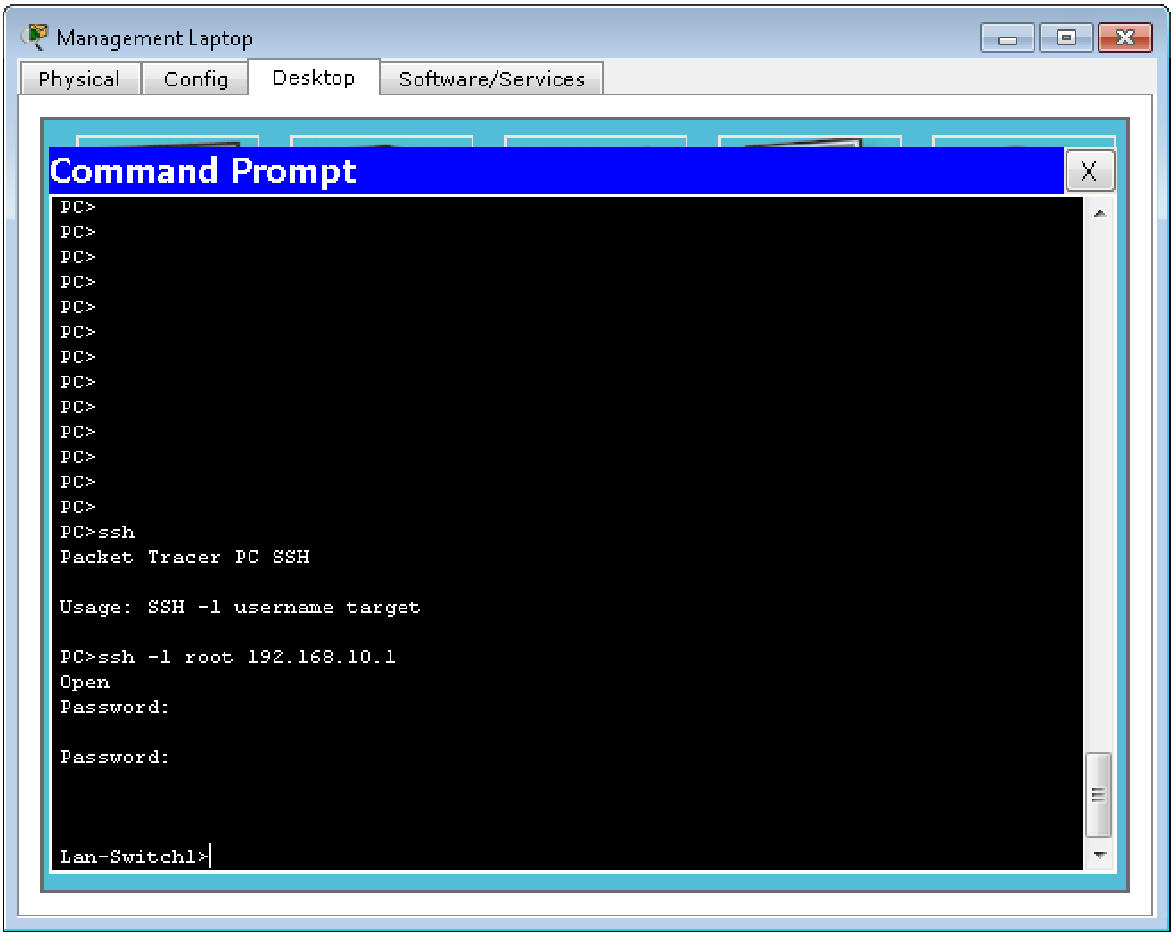
```
Lan-Switch1(config)#crypto key generate rsa
The name for the keys will be: Lan-Switch1.test.lyit.ie
Choose the size of the key modulus in the range of 360 to 2048 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [512]:
% Generating 512 bit RSA keys, keys will be non-exportable...[OK]

Lan-Switch1(config)#
```

In a real deployment, we will use keys bigger than 512. The key size should be based on policy rather than randomly chosen during installation! Keys may take some time to generate. For this lab, use 512.

If you now carry out the command **show crypto key mypubkey rsa** again you can see the key in hexadecimal.



```
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>
PC>ssh
Packet Tracer PC SSH

Usage: SSH -l username target

PC>ssh -l root 192.168.10.1
Open
Password:

Password:


Lan-Switch1>
```

Now go back and try to login from Laptop1. This time use the ssh command and the "MyTelnet" password.

You will still need to use enable and the secret password (MyPassword) to get to the elevated command prompt.

SSH Security

Even with SSH, we are vulnerable to brute force attacks. Linux by default does not alter SSH behaviour after repeated password failures. This is bad! Additional tools are required to enforce the desired behaviour.

With a Cisco switch or router, same problem! Make sure you use the options to the login command which introduce delays on failed password attempts.

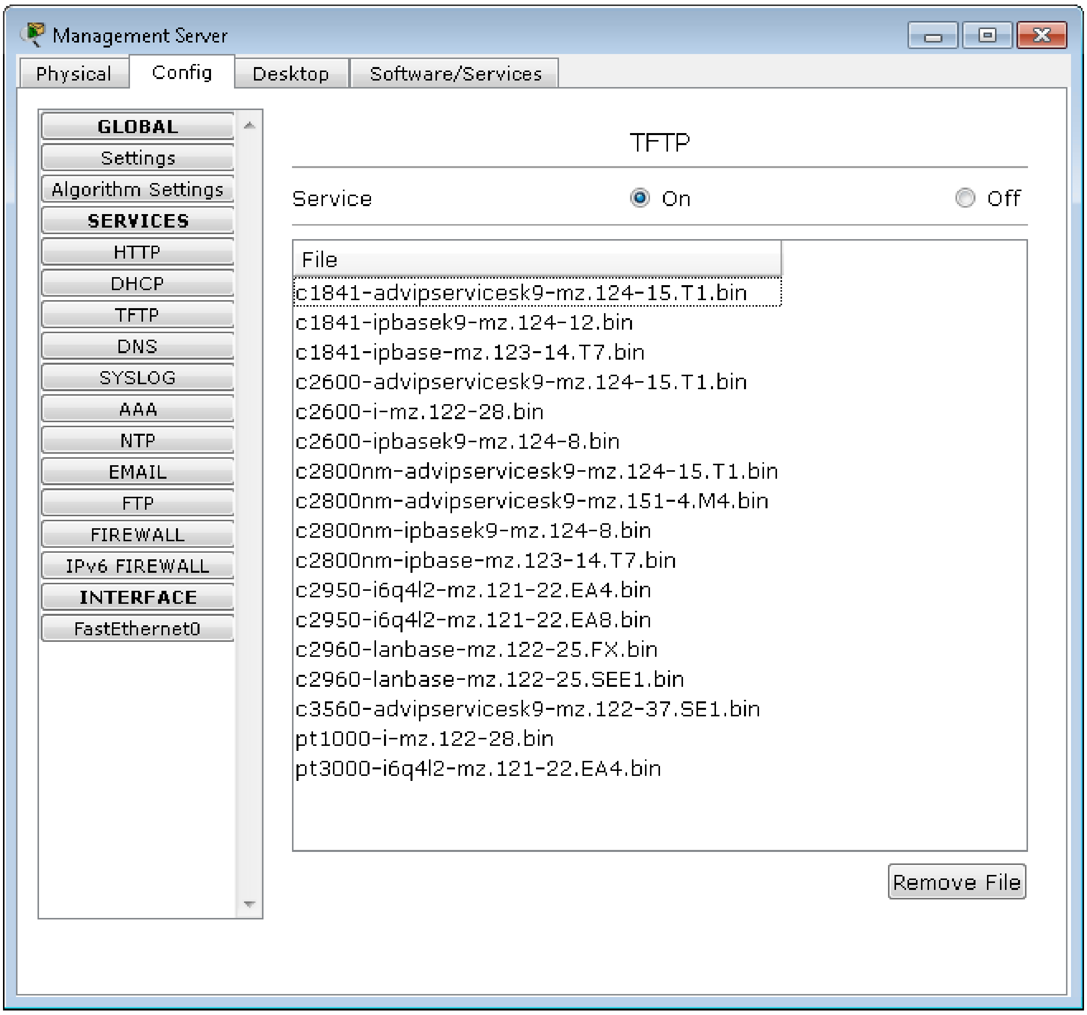
TFTP

The final service we are going to consider is Trivial File Transfer Protocol, defined by RFC1350. This was designed as a cut-down version of FTP, with no authentication or security. It is used as a standard by most networking equipment to copy images and configurations up and down from equipment. It can be implemented in a very small amount of memory and is very economic on resources.

When we set up a network for the first time, we almost always need to implement a TFTP server to get and set configurations, firmware updates and software onto equipment. TFTP is one of those odd protocols that may not be well supported. For example, many of us have given up on using TFTP on Debian/Ubuntu.

Using TFTP

On the management server, go to the TFTP service. As you can see, there are firmware files for many switch and routers types already saved in the server and the server is “on”.



TFTP

Service ● On ○ Off

File
c1841-advipservicesk9-mz.124-15.T1.bin
c1841-ipbasek9-mz.124-12.bin
c1841-ipbase-mz.123-14.T7.bin
c2600-advipservicesk9-mz.124-15.T1.bin
c2600-i-mz.122-28.bin
c2600-ipbasek9-mz.124-8.bin
c2800nm-advipservicesk9-mz.124-15.T1.bin
c2800nm-advipservicesk9-mz.151-4.M4.bin
c2800nm-ipbasek9-mz.124-8.bin
c2800nm-ipbase-mz.123-14.T7.bin
c2950-i6q4l2-mz.121-22.EA4.bin
c2950-i6q4l2-mz.121-22.EA8.bin
c2960-lanbase-mz.122-25.FX.bin
c2960-lanbase-mz.122-25.SEE1.bin
c3560-advipservicesk9-mz.122-37.SE1.bin
pt1000-i-mz.122-28.bin
pt3000-i6q4l2-mz.121-22.EA4.bin

Go to Lan-Switch1 and execute the command **show file systems**. This should show you two locations, flash and NVRAM. Now check what files are in each using the commands **dir flash:** and **dir:nvram**.

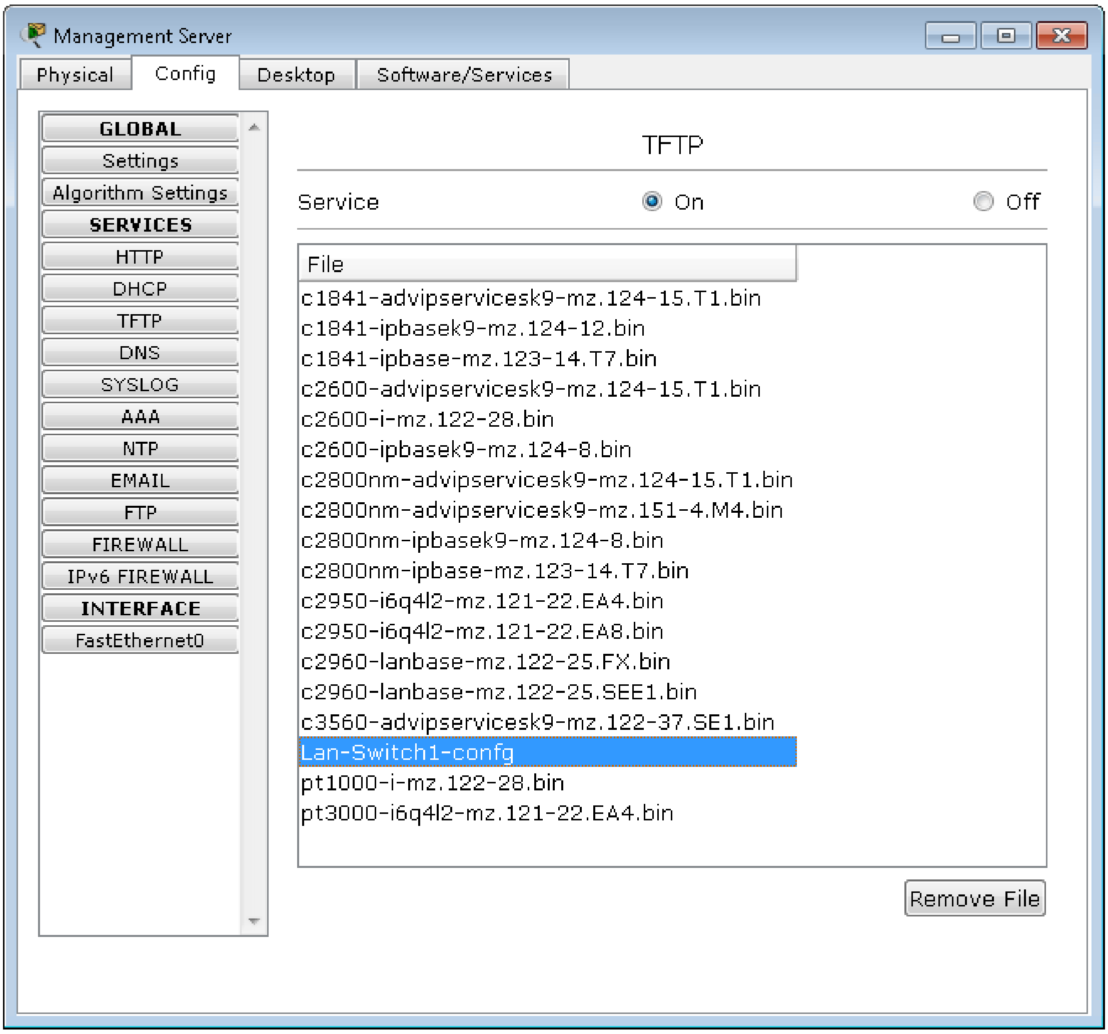
First ask, backup the running configuration to the tftp server.

```
Lan-Switch1#copy running-config tftp
Address or name of remote host []? 192.168.10.254
Destination filename [Lan-Switch1-confg]?

Writing running-config...!!
[OK - 1877 bytes]

1877 bytes copied in 0 secs
Lan-Switch1#
```

Check the management server, the configuration switch should be there.



TFTP

Service ● On ○ Off

```
File
c1841-advipservicesk9-mz.124-15.T1.bin
c1841-ipbasek9-mz.124-12.bin
c1841-ipbase-mz.123-14.T7.bin
c2600-advipservicesk9-mz.124-15.T1.bin
c2600-i-mz.122-28.bin
c2600-ipbasek9-mz.124-8.bin
c2800nm-advipservicesk9-mz.124-15.T1.bin
c2800nm-advipservicesk9-mz.151-4.M4.bin
c2800nm-ipbasek9-mz.124-8.bin
c2800nm-ipbase-mz.123-14.T7.bin
c2950-i6q4l2-mz.121-22.EA4.bin
c2950-i6q4l2-mz.121-22.EA8.bin
c2960-lanbase-mz.122-25.FX.bin
c2960-lanbase-mz.122-25.SEE1.bin
c3560-advipservicesk9-mz.122-37.SE1.bin
Lan-Switch1-confg
pt1000-i-mz.122-28.bin
pt3000-i6q4l2-mz.121-22.EA4.bin
```

Now if you are sure you have done this correctly, wipe the switch. Use the command **write erase** followed by **reload**. Do a **show run** to make sure your configuration is really gone. Your switch is just called “switch” again.

We need to be able to communicate with the TFTPserver in the simplest way to get the configuration back. Leave VLAN1 as the default for everything and set an IP address on it. Add the address 192.168.10.1/24 to VLAN1 and remember to do a **no shut** command. Try to ping the Management Server to make sure you have got this right.

Next thing to do, copy the saved configuration on the TFTP server back to the running configuration.

```
Switch#copy tftp running-config
Address or name of remote host []? 192.168.10.254
Source filename []? Lan-Switch1-confg
Destination filename [running-config]?

Accessing tftp://192.168.10.254/Lan-Switch1-confg...
Loading Lan-Switch1-confg from 192.168.10.254: !
[OK - 1877 bytes]

1877 bytes copied in 0 secs
Lan-Switch1#
```

Remember to do a **write memory or copy run start** command, as you need to save the running configuration before you restart.

In a real network, we will use TFTP servers like this when we are doing upgrades. When you ran the command **dir flash:** you saw a file called “c3560-advipservicesk9-mz.122-37.SE1.bin” or similar. This is the system firmware. Periodically, as new firmware is released, updates will be TFTP’d down to the switch.

This was a very simple example. In a real network, we could do configuration management like this, but we would need to find some way to script it. This is why so many network engineers have learned to use Python.