Before Clouds:
Hardening FTP

A cloud está na moda mas já se transferiam e partilhavam ficheiros antes do aparecimento da Cloud.

Esta apresentação destina-se a demonstrar as evoluções do protocolo FTP e como se pode aumentar a segurança das transferências de ficheiros e dos sistemas em geral

Before Clouds: Hardening FTP

- Agenda:
  - FTP, o que é?
  - Porquê usar?
  - Porquê não usar?
  - FTP no século XXI
  - Hands-on com ProFTPd

Nesta apresentação vamos ver o que é o protocolo FTP, porque é que ainda poderá fazer sentido usá-lo hoje em dia, porque é que poderá fazer sentido não o usar, como é que pode ser configurado para que se mantenha uma opção válida no século XXI e iremos fazer algumas demonstrações.

FTP, o que é?

File Transfer Protocol

O nome diz tudo: Protocolo para transferência de ficheiros.

Torna-se um protocolo interessante porque implementa mecanismos de retoma de transferências interrompidas

FTP, o que é?

1985

RFC 959, Outubro de 1985

Mas, após 30 anos, fará sentido considerar o protocolo obsoleto?

FTP, o que é?

Conteúdo circula em “*plain*”
(que é como quem diz: **sem cifra**)

Na sua definição inicial, toda a comunicação FTP (controlo e dados propriamente ditos) é feita sem recurso a qualquer mecanismo de cifra, o que implica que todo e qualquer activo de rede existente entre o cliente (aplicação que se liga a um servidor) e o servidor poderá interceptar e até mesmo adulterar a comunicação, credenciais inclusive.

Porquê usar?

"Clientes" e comandos **simples**

Desde a utilização mais clássica do comando FTP até aos interfaces ortodoxos de gestão de ficheiros (onde existem 2 painéis: de um lado os ficheiros locais do sistema cliente, do outro os ficheiros existentes no servidor – estilo Midnight Commander, Norton Commander, entre outros), a utilização acabava por ser relativamente simples.

Porquê usar?

Implementações em inúmeras
linguagens de programação

Sendo um protocolo sobejamente conhecido, este encontra-se implementado em praticamente todas as linguagens de programação, o que quer dizer que bastará fazer uso de um módulo/uma biblioteca já existente para que uma aplicação informática seja capaz de recorrer ao protocolo FTP para transmitir ou receber ficheiros.

Porque **não** usar?

Particularidades ao nível de **portos**

Possivelmente uma das maiores complexidades protocolares prende-se com os portos TCP usados numa comunicação FTP. De uma maneira geral os utilizadores identificam o porto 21 como “o porto do FTP” mas, na verdade, este só é usado para controlo (autenticação, por exemplo). A transferência de dados (ficheiros) em si é efectuada noutros portos (se a transferência for em modo activo, é usado o porto 20; em passiva é alocado um conjunto de portos e a comunicação de controlo é que se encarrega de identificar qual irá ser usado).

Esta questão coloca entraves à configuração de activos de rede e uma alteração da configuração do servidor que não tenha em conta as rotas abertas pode causar problemas.

Porque **não** usar?

Inseguro para ser usado em ambiente não controlado

Em virtude de toda a comunicação se realizar sem qualquer tipo de cifra, o protocolo FTP é inseguro e deve ser evitado em comunicações onde não se tem conhecimento e controlo de todos os activos de rede.

Adicionalmente, a configuração por omissão da maioria dos servidores de FTP valida as credenciais recorrendo aos utilizadores de sistema, o que quer dizer que se estão a expor dados que poderão permitir acesso a outros serviços e, no limite, acesso total ao sistema.

FTP no século XXI

É mesmo necessário utilizador de sistema?

PORTO LINUX

Devemos questionar-nos se temos realmente a necessidade de que a validação de credenciais seja realmente feita contra utilizadores de sistema.

Se tal não for necessário, então a validação deve ser feita por intermédio de um ficheiro “passwd” específico para o efeito ou base de dados.

Se o serviço foi inicialmente configurado e facultado com recurso a utilizadores de sistema e, a certo ponto, pretende-se alterar apenas para utilizador FTP, é possível extrair dos ficheiros de sistema a informação necessária por forma a não ser necessária a atribuição de novas credenciais (verificar o ficheiro scripts/system_to_proftpd.sh).

FTP no século XXI

Segurança: **FTPS** ou **SFTP**?

Ao nível de segurança da comunicação, surgiram duas evoluções para o FTP:

- O FTPS está para o FTP como o HTTPS para o HTTP, ou seja, recorre a SSL e, como tal, necessita de ter um certificado, preferencialmente assinado por uma CA reconhecida, para evitar avisos; a desvantagem do FTPS é que mantém a complexidade de utilização dos portos do FTP:
- O SFTP é um protocolo sobre SSH e usa apenas um porto (por omissão o 22) mas uma vez que não há certificado nem CA, é conveniente que a chave pública e o “fingerprint” do servidor estejam disponíveis em sítios confiáveis, para que os utilizadores possam ter garantias que se estão a ligar aos servidores que efectivamente pretendem contactar.

Hands-on com ProFTPd

Autenticar utilizadores através de ficheiro próprio

A autenticação FTP com recurso a um ficheiro próprio com o formato do /etc/passwd permite:

- Identificar rapidamente quem tem acesso a cada um dos serviços de FTP;
- Assegurar que não há o risco de um utilizador FTP ganhar acesso, com o seu utilizador, ao sistema;
- Virtualizar os utilizadores: temos vários utilizadores de um serviço FTP mas, ao nível do sistema, os ficheiros são todos da “ownership” de um único utilizador de sistema (exemplo: uma entidade tem várias pessoas que podem fazer operações sobre ficheiros mas pretendemos que, aos olhos do sistema, o utilizador represente a entidade)

Ficheiros associados:

- etc/proftpd.conf.1.no_sys_users
- scripts/system_to_proftpd.sh

Hands-on com ProFTPd

Assegurar a utilização de SFTP

A utilização de SFTP requer:

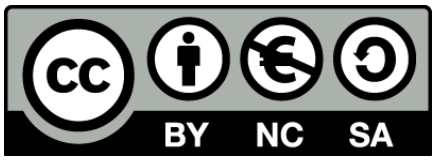
- Utilização de chaves RSA ou DSA (neste caso contamos ter ambas para suportar a maior quantidade possível de clientes)
- Se tivermos SSH a correr no porto 22 da máquina, deve-se configurar um outro porto para SFTP mas é conveniente que, para o mundo, o servidor de FTP esteja no porto “standard” (22) e o de SSH, se efectivamente necessário, num outro porto (não-standard, de preferência fora da gama de “well known ports”)

Ficheiros associados:

- etc/proftpd.conf.2.sftp
- scripts/create_sftp_keys.sh
- scripts/sftp_test.sh

Q&A

Manuel Silva



PORTO LINUX

Estou disponível para quaisquer questões relacionadas com esta apresentação através do endereço:
msilva@portolinux.net

Obrigado pela atenção dispensada.